Refining a Team Training Approach to Prepare Command Staff for Wargaming

Randy Jensen[1], Evan Finnigan[1], Grace Teo[2], Lisa Townsend[3]
Stottler Henke Associates, Inc.[1], Quantum Improvements Consulting, LLC[2],
U.S. Army Combat Capabilities Development Command – Soldier Center (CCDC – SC)[3]

INTRODUCTION

This paper gives an update on the design and implementation of a team trainer for preparing command staff for the collaborative processes of course of action (COA) development and wargaming. The target training audience are command staff at battalion and higher echelons, who must bring together the expertise of different warfighting functions as they synchronize plans in a key stage of the Army's military decision making process (MDMP). This process requires staff members to understand the interrelationships between roles and functions, which relates to individual and team-related training objectives for participants to effectively gain and use knowledge at three levels: within their own roles, provided to other roles, and needed from other roles.

For this team training application, the technical approach has evolved over several iterations in terms of the training use case and target audience, the role of human instructors driven by the use case, the assessment model and methods, and the plans for standards-oriented implementation methods. The earliest stage involved analysis of the relevant team dimensions, resulting in a model for teamwork constructs applicable to wargaming performance assessment (Teo et al, 2021). Also initial prototyping included preliminary design of a standards-oriented implementation (Jensen et al, 2021) with the structure for a distributed team trainer and mechanisms making use of the GIFT architecture (Generalized Intelligent Framework for Tutoring, Sottilare et al, 2012). An initial prototype version of the training application was presented in a previous GIFTSym paper (Jensen et al, 2022), with examples of team assessment methods in the context of wargaming exercises. Initial feedback from instructors led to a modified prototype design to support a use case adapted for a specific need in current wargaming training. Where the initial prototype attempted to create an approximation of a wargaming exercise in a distributed trainer, the modified design aims more toward ***preparing*** staff for wargaming, rather than ***conducting*** wargaming. This is because an important prerequisite for being able to conduct wargaming effectively is the understanding of the team roles. When the staff team has a shared understanding of the various team roles, they would be able to provide better information in support of each other and preemptively avoid errors when presented with an operating situation (Salas et al, 2014). Hence, the priority became targeting a need for training staff to first become familiar with the process of cross-functional coordination with different roles, and how they work together to synchronize plans for a COA. This prepares the staff for the next step where the focus moves more heavily to scenario-specific wargaming decisions. In order to support this modified exercise design and training objectives, a second prototype has been developed.

This paper gives an overview of the initial implementation of the second prototype, as context for a discussion of planned directions for conforming and integrating with GIFT. Although the earliest design outlined planned methods for building the system based on the GIFT architecture, the use case has changed and shifted away from earlier plans for using GIFT artifacts like the Domain Knowledge File (DKF). For example, as instructors elaborated on the objectives and practices for wargaming preparation exercises, they described team feedback methods that do not readily fit the mold for the kinds of performance scoring mechanisms that may be used for other domains. The team competencies that instructors reference in this domain tend to be limited in number and expressed at a high level. In practice, the more nuanced elements of team competencies are covered informally in after action review through instructor-led discussion and team self-reflection. However, there is structure provided by the model for team dimensions in wargaming, and this is the starting point for revisiting how this application can be integrated with GIFT and benefit from its conventions. This paper outlines considerations for how exercise management, data flow, and assessment can be constructed with GIFT for this application. These considerations also may ultimately inform more general practices for other similar collaborative team decision-making domains. For instance, the GIFT DKF construct may be used to collect instances of teamwork markers generated by either the instructor or the automated rules in the environment. An example of where a marker is created is when a participant identifies information relevant to the COA that can be supplied by another role, which is an indicator of team cognition.

The following topics are discussed in this paper:

- Training application overview with example team learning objectives
- Architecture and planned interoperability with GIFT
- Assessment methods

TRAINING APPLICATION OVERVIEW

As discussed above, the application described in this paper aims to provide team familiarization exercises to train U.S. Army command staff at battalion and higher echelons in the team processes associated with COA development, analysis, and wargaming. The prototype is under development as part of an effort called Reusable Automated Assessment and Feedback for Teams (RAAFT), which is being conducted for the U.S. Army Combat Capabilities Development Command – Soldier Center (CCDC – SC).

Training Environment

The RAAFT prototype is constructed as a distributed, browser-based synchronous team trainer, which allows participants to be either remote or co-located. Exercises are led by a human instructor who also plays the part of the Lead planner, and participants are each assigned to one of six command staff roles representing different warfighting functions – Intel, Movement & Maneuver (MM), Indirect Fire (Fires), Aviation (AV), Protection (PRO), and Sustainment (SUS). Participants initially review pre-briefing materials about the operational scenario, such as a tactical map, mission statement, and enemy order of battle. As a familiarization exercise, when it comes to the consideration of a specific scenario and COA, one of the overarching goals is to orient the training audience more toward simply asking the right questions (process), as opposed to trying to arrive at the best decisions (performance outcomes). The purpose is to encourage thinking about the contributions of different warfighting functions, and also keep the exercise overhead to a minimum in terms of the amount of scenario-specific knowledge that participants must consume.

The exercise itself is conducted mostly by collaborating in a shared message panel, where the Lead / Instructor posts Prompts to initiate staff discussion on different Topics. The Lead / Instructor has a prescribed list of Topics with associated Prompts that can be sent to the message panel, but these Topics are only visible to the Lead / Instructor, and not to the training audience. Figure 1 below shows the Lead / Instructor screen, which includes the panel with instructor tools that are unique to this role (lower left, with a list of Topics and a Prompt field). The map panel and message panel are common to all participants, who engage in discussions triggered by Prompts, by either replying inline on existing threaded discussions, or adding new messages (using the input box) which are treated as new threads. Since the training environment is intended to support a variety of settings, participants may also communicate by other means (even verbally if in close proximity), but the training application has no access to such interactions. As participants type messages, a selection of auto-complete options is available based on their partial inputs, or they can proceed to just type free text. Since the auto-completes come from a predefined library with markup for their meaning and relevance to Topics, this is one avenue for the system to understand messages coming from participants, and apply automated rules. For free-text inputs, most of the burden of understanding is on the Lead / Instructor. In the exercise snapshot below, the Lead / Instructor has sent two Prompts (seen in the message panel), most recently leading to discussion on the current active Topic "Indicators for EN repositioning." In the Instructor tools panel (bottom left) the current Topic is marked with a check, to reflect the judgment that the staff team has adequately covered the Topic.

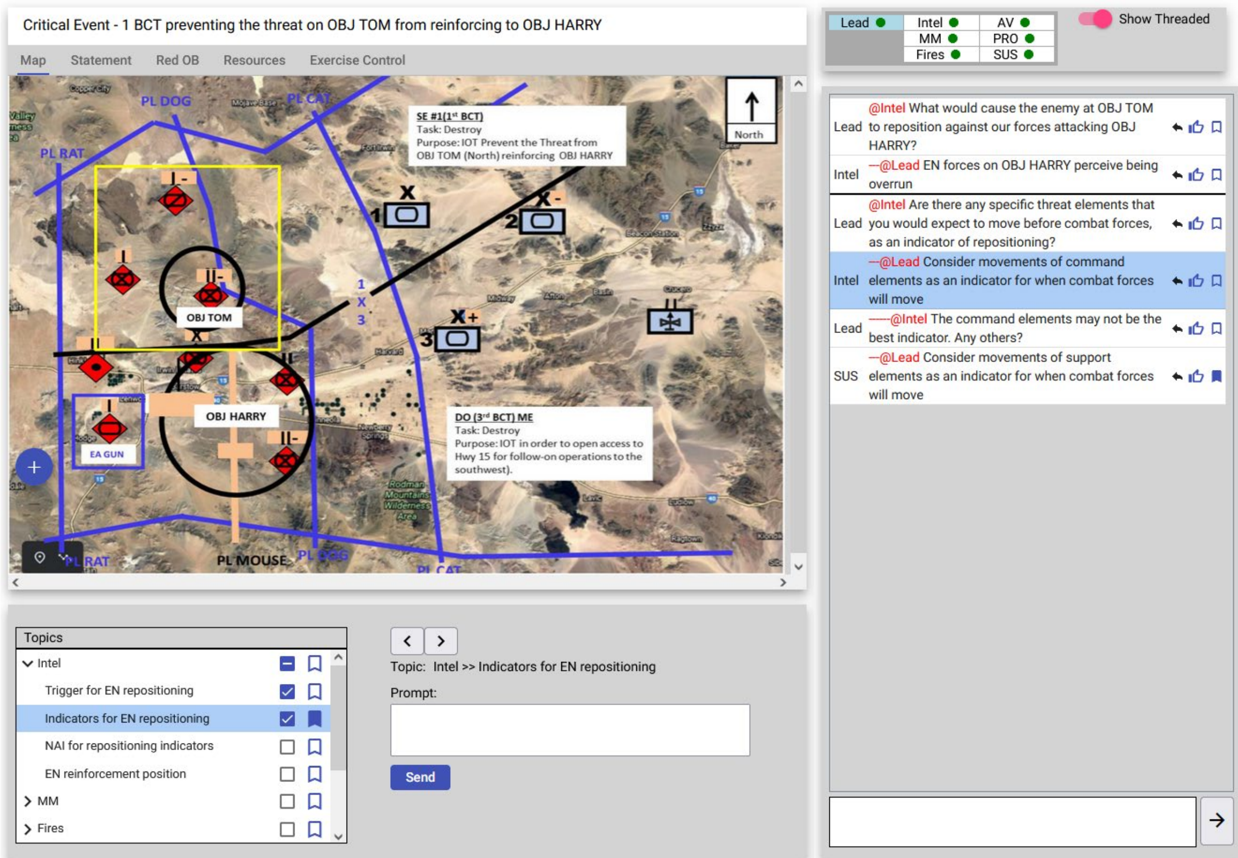

Figure 1. Training Application View of Lead / Instructor Screen During Exercise

Instructor Prompts are typically directed to a specific role, but anyone can take part in the discussion at any time. This is often essential, as the assigned roles in the exercise may not necessarily correspond to real-world expertise, sometimes by intention, so cross-functional participation is encouraged and in fact fundamental to effective wargaming processes.

As the exercise unfolds, noteworthy examples of teamwork (either positive or negative) are identified with markers. Markers can be created either by the Lead / Instructor or by automated rules in the system, and can be associated with a specific message and/or a Topic and/or a more general observation. In the example above, there is a marker (small bookmark icon) that has been created and associated both with the current Topic ("Indicators for EN repositioning") and also with the most recent message from Sustainment ("Consider movements of support elements…"). The markers capture significant examples of teamwork from the exercise for further discussion in after action review (AAR). In this case, the marker was created by an automated system rule. The analysis is made possible by the fact that the participant in the Sustainment role used an auto-complete message, which could be recognized by the system for its relevance to the current Topic. Two operations happened as a result. First, from internal associations authored in the auto-complete library, the system determines that the message is considered an optimal response for the current Topic. Specifically, the enemy support elements are an effective indicator that maneuver units are repositioning, and this is a better choice than the initial response from Intel (which refers to enemy command elements). So the Topic is automatically marked as covered in this case, although more generally it can be either the system or the Lead / Instructor making judgments about when a Topic is adequately covered. The second operation is the creation of a marker based on observed teamwork processes. Although the original Topic involved a Prompt that was directed to Intel, it was the Sustainment role that contributed the optimal response. This is one situation that the system identifies as positive teamwork, so it creates a marker tagged with the Supporting Behavior team dimension. Using the popup window below in Figure 2, the Lead / Instructor reviews the marker and adds further information.

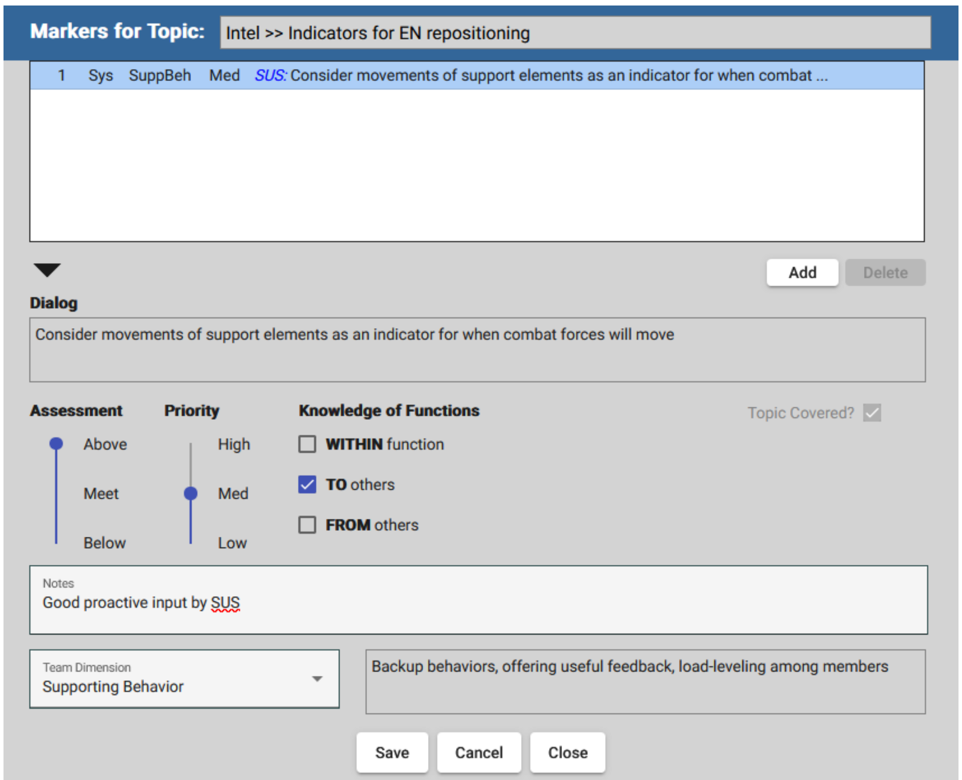

Figure 2. Marker Review Popup for Lead / Instructor

The representation for markers supports several kinds of supplemental information, including:

- Contextual information
    - Relevant message and Topic, if any
- Qualitative assessment
    - Above, Meet, or Below expectations
- Priority
    - High, Med, or Low, for reference when preparing AAR
- Functional knowledge
    - Concept relating to whether the marker relates to a particular participant's knowledge WITHIN their own functional area, to be provided TO another function, or needed FROM another function

- Relevant team dimension
  - Referencing a model of team dimensions applicable to this domain, such as Team Cognition, Information Exchange, and Supporting Behavior (Teo et al, 2021)
- Additional notes from the Lead / Instructor

The Lead / Instructor input in this example includes an Assessment notation of Above expectation, and a Knowledge of Functions notation that this exchange involved information flow To others (in this case, Sustainment knowledge about logistics elements, provided to Intel and the rest of the staff). The Lead / Instructor also adds a comment that this is good proactive input.

Learning Objectives

The learning objectives targeted by the RAAFT training application are mostly associated with team processes. In contrast to team performance which places more emphasis on outcomes, team processes relate to the dynamics of interactions or cognitive states within team members (Grand et al, 2013), and are foremost in this application since a major exercise goal is to familiarize participants with the different staff roles and their cross-functional dependencies. There are several team processes activated in the training application, and so one of the design questions for structuring a competency model for this domain involves defining how competencies will be measured and retained and tracked over multiple exercise iterations, and potentially within and across teams. Specifically, we consider four different kinds of measures:

- **Topic coverage.** For the Topics associated with the operational scenario in the exercise, did the team adequately cover the questions that needed to be considered or addressed? This corresponds directly to existing practices followed by instructors when they monitor wargaming exercises and maintain a checklist to mark notes organized by topics in a mental model tailored for the scenario. In this sense, each Topic in the exercise has an associated learning objective that can go into a competency model (Trigger for EN repositioning, Indicators for EN repositioning, NAI for repositioning indicators, EN reinforcement position…).
  - Example from above: did the team adequately identify considerations about the indicators for enemy repositioning?

- **Team dimensional constructs.** For teamwork processes exhibited in the exercise, what are the relevant team dimensions? This makes reference to a model for teamwork constructs applicable to wargaming performance assessment developed earlier in this effort (Teo et al, 2021). However, it is challenging to treat the team dimensions as competencies in the conventional way that other competencies are handled, because of their abstract nature. It remains a research question whether and how it is suitable to construct an exercise that would aim to mark an individual or team as having reached a certain scored proficiency at Team Cognition or other similar measures. Instead, the team dimensions are treated as supplemental information to accompany markers for more concrete measures such as the instances of Topics covered well or poorly. Thus the initial implementation treats the team dimensional constructs as having a secondary role rather than being independent competencies tied to learning objectives.
  - Example from above: what team dimensions were exhibited in the process of the staff's discussion of indicators for enemy repositioning?

- **Post-exercise self-reporting.** As participants reflect on the exercise during AAR, what did they learn about their own roles and others? This kind of explicit reflection (often called "sustains and improves") is a common practice with existing training, so it is mirrored in exercises to be conducted in this environment. Since an AAR is conducted within the training environment, where

participants can refer back to the discussion of different Topics, self-reported takeaways can be compared with markers from the Lead / Instructor and also add enhanced information.
  - Example from above: does the participant playing the Intel role echo anything they learned from the Sustainment suggestion to consider enemy support elements rather than command elements?

- **Specific performance measures.** Aside from other measures above, what are concrete indications of team processes that can be inferred from analysis of exercise data? Examples include communication dynamics (e.g., statistics about relative levels of participation from different roles), time to complete Topics, repetition or disjointed communication across threads.
  - Example from above: How long did the staff spend on the discussion of indicators for enemy repositioning?

ARCHITECTURE AND PLANNED INTEROPERABILITY WITH GIFT

At the highest level, the RAAFT prototype is a server-based team training application that users must access synchronously from their browsers. Exercises require users to take unique roles corresponding to the different staff warfighting functions, so the process for joining a training exercise must include the assignment of roles. For the purposes of GIFT integration, the RAAFT prototype is treated as an external training application, which is readily supported in an architecture where a GIFT Cloud instance communicates with the RAAFT application and its server. This section describes the architecture in more detail, starting with the structure of the training application itself, and then the integration and data flow with GIFT. The architecture for interoperability with GIFT is currently under development, so the discussion of integration methods is aimed at the current design concept.

RAAFT Training Application

The RAAFT application architecture takes the form of a MEAN architecture which stands for MongoDB, Express, AngularJS, and Node.js. This is a standard architecture used to build web applications.

RAAFT Client

The RAAFT client is an Angular application. Angular is a front-end web framework designed for creating single-page web applications. Angular is created on top of Node.js which is a JavaScript runtime for building server-side or desktop applications. The RAAFT client is a thin interface that delivers information and takes user interaction but leaves all significant processing to the server. Figure 1 in the previous section shows the client interface as seen by the Lead / Instructor.

RAAFT Server

The server is a Node application that provides a REST API (Representational State Transfer Application Programming Interface) for the web client application. This REST API manages exercise data flow to and from users, including the delivery of scenario information and the recording of all user inputs including exercise-related communications as well as actions by the Lead / Instructor such as the use of teamwork markers. One of the key requirements of a REST API is that it is stateless, which means that the server does not store any information related to previous requests. All data that needs to be persisted is stored in the database. The REST API receives all data in JSON (JavaScript Object Notation) format. This is a natural choice for a MEAN stack because JavaScript and TypeScript objects can be easily serialized into JSON, and the MongoDB database can easily process data in JSON format for storage.

In addition to receiving user-created data and persisting it to the database, the RAAFT application server also performs a range of processing tasks. The RAAFT application server can use rules to create assessments and also do some basic processing to manage the session. For example, when the Lead role ends a session, the server clears all of the messages, the currently selected topic, markers, and topic statuses.

Database

A MongoDB database stores and persists all application-related data. A MongoDB database uses a non-relational approach to storing data with flexible data models. The MongoDB database stores all of the dynamic exercise data: messages, status of Topics, and markers as well as the static data like the Prompts and the full library of auto-complete message options. All of this static data is pulled into the database from JSON files at the beginning of the exercise, so content is added or modified via those JSON files. An export capability is supported in order to save a log of the dynamic exercise data at the end of an exercise.

Deployment

The RAAFT client, server, and database software components all need to run simultaneously, so they are containerized using Docker, which packages software and execution requirements into a lightweight environment to alleviate the need for separate manual installation. The three software components are deployed on a Linux server running Docker. The Linux server is also configured to accept requests on specified ports to allow users to request the web application and then to allow the web application to connect to the REST API to receive data.

Interoperability with GIFT

Figure 3 below shows a simplified architecture for the RAAFT training application and GIFT.

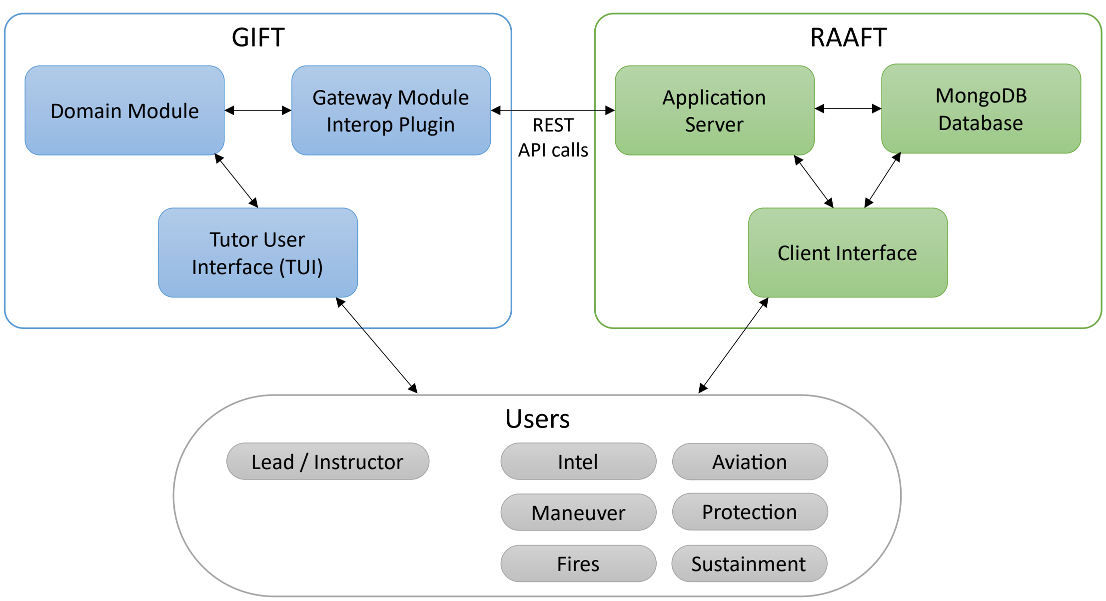

Figure 3. Interoperable Architecture with the RAAFT Training Application and GIFT

All users interact with both applications through a browser, with simultaneous active sessions to access (i) a GIFT Cloud instance through the Tutor User Interface, and (ii) the training application through the

RAAFT Client. RAAFT and GIFT communications are managed with REST API calls between the RAAFT Server and the GIFT Gateway Module Interop Plugin. For the current implementation efforts, the focus is on interoperability within the scope of a single exercise. Although the mechanism is designed to be in place to maintain records of exercise results and team competencies in GIFT, the initial implementation does not make use of GIFT capabilities to manage exercise sequencing or instructional models.

The initial integration between the training application and GIFT consists of two parts: coordinating the launch of an exercise with users assigned to specific roles, and delivering training application state messages to the GIFT Domain Module.

Exercise Launch

The goal for an ideal configuration is to make the exercise launch process as seamless as possible, where an exercise participant can go from a GIFT lobby straight to participation in the training application, with robust support for loss of connectivity to either application (the GIFT Cloud instance or the RAAFT training application) without loss of session role assignments and data. The initial architecture reflects a step toward this goal, but with the requirement that users open independent browser instances for both applications. In this configuration, the sequence is as follows:

Table 1. Exercise Launch Sequence

Event	Role	GIFT	RAAFT
Pre-launch	All users (including the Lead / Instructor as well as all participants taking command staff roles)	Lobby interface open in browser, showing a page to select a role for the exercise. The Lead / Instructor has unique credentials for this step, as the only person authorized to select that role. All others can only select from the command staff roles.	Client open in browser, showing a blank landing page confirming successful connection, and awaiting role selection.
Role assignment	Any user	Upon selecting a role, the Gateway Module Interop Plugin sends a configuration message to the RAAFT server, associating the user with a role.	After receiving the configuration message from GIFT, the landing page shows the role assignment. Until the exercise starts, all participants see their roles, but do not enter the exercise environment yet.
Exercise start	Lead / Instructor		The client interface for the Lead / Instructor includes exercise control tools. When the Lead / Instructor starts the exercise, the exercise environment becomes active in the client interfaces for all participants. Any participant joining after this step goes directly from role assignment to the active exercise.

Data Flow

GIFT treats RAAFT as an external training application in this architecture, in the sense that assessment logic is internal to the RAAFT application, and the assessment outputs are primarily the teamwork markers created either/both by the system and the human Lead / Instructor role. Fundamentally all data flow between the training application and GIFT before, during, and after an exercise takes place via REST API calls. This approach matches well with the existing implementation of the RAAFT prototype which already makes use of a REST API. For the purposes of GIFT integration, the REST API is implemented inside of a GIFT Gateway Module Interop Plugin, which then allows Java code written on the RAAFT side to interact with Condition Classes in the GIFT Domain Module. During an exercise, the updates to Topic status and the Teamwork markers exercise are conveyed to GIFT by this mechanism. Both of these types of data are serialized as JSONs using a schema with the necessary data.

For markers originating from humans, the data flow will likely remain primarily a relay function as these markers are sent as state messages to a GIFT Condition Class. There is also the possibility of using or adapting the Observed Assessment mechanism in the GIFT Game Master tool as a means for the Lead / Instructor to input markers, as long as this doesn't create confusion switching applications during the exercise. However, for system generated markers, there is logic for rules that process exercise data to identify situations where certain markers can be created and tagged. Although the initial plan in terms of interoperability is for this logic to remain internal on the RAAFT Server, it is a future goal to experiment with abstraction by implementing at least this portion of assessment logic to an implementation within GIFT Condition Classes associated with nodes in the DKF concept hierarchy. This next phase of integration requires dynamic exercise data such as Prompts and participant messages to be sent during the exercise. In addition, the Condition Classes need to be initialized with scenario knowledge consisting of static data for Prompts, Topics, and the auto-complete library. The following section goes into further detail about assessment methods in the training application, and the relationship to GIFT mechanisms for assessment via the Domain Module.

ASSESSMENT METHODS

As discussed above, the concept for the implementation of assessment methods is phased. The plan for a future phase is to explore abstracting assessment logic to run using GIFT constructs external to the training application. However, initially, from a GIFT integration standpoint, RAAFT is considered an existing training application with its own internal assessment engine. In this configuration, the markers are treated as the primary training application state messages passed from RAAFT to GIFT, and the markers are collected by a Condition Class in the GIFT Domain Module. DKFs on the GIFT side associate incoming marker data with concept hierarchy nodes, which predominantly correspond to the Topics in the exercise.

Referring back to the view of the training environment in Figure 1, the hierarchy of Topics is organized by warfighting functions corresponding to the staff roles occupied by participants. Under the Intel role, four topics are visible: Trigger for EN repositioning, Indicators for EN repositioning, NAI for repositioning indicators, and EN reinforcement position. Each of these Topics (and all Topics associated with other roles) has a counterpart in the concept hierarchy in GIFT. When the team of participants covers a Topic, with any associated markers created to convey supplemental information such as relevant team dimensions or the relationship to knowledge of functions, the delivery of a message to GIFT becomes the basis for a record of the results for the corresponding concept.

As a collaborative team problem-solving domain, it is difficult to define a set of competencies that matches the scope of instructional training objectives. Both for the broad areas of wargaming and MDMP, and the more specific wargaming preparation and team familiarization exercise delivered by the RAAFT training application, instructional objectives are mainly oriented at giving cadres of exercise participants opportunities to practice team collaborative processes. Conventionally implementable artifacts like enumerated competencies with scored levels of mastery tend to be only indirectly used in existing training for these domains, and only at a high level (example: ELO-AOC-12.1 "Use the Military Decision-making Process (MDMP) to plan a tactical operation"). Since team compositions routinely shift from one exercise to another, there is little meaning to an approach that would involve persistent team-level models of mastery. If a team of six individuals demonstrates effective teamwork processes in the handling of Intel related topics in an exercise, that may or may not be an indicator of likely teamwork in a future exercise to involving four of those six, with two roles swapped out. Another approach as an alternative to a persistent team model for a specific team composition might be to simply distribute findings about teamwork competencies to each participant, stored in their individual student models. But this approach also has downsides, because there are many situations where it isn't appropriate to attribute positive and negative team processes to each individual on the team.

A further complication comes from the fact that teamwork inherently relates to processes or situations that may arise from differences across participants or roles. Referring back to the exercise example shown in Figure 1, the situation in this case started with the Intel role initially suggesting a sub-optimal response to a Prompt (to consider enemy command elements as an indicator of repositioning). From a straightforward assessment perspective, this response from Intel could be considered a negative training point, perhaps an individual error. However, the emphasis of the exercise is on teamwork processes, as opposed to performance outcomes. And the process in the same situation involved a follow-up contribution from Sustainment, suggesting a better response (to consider enemy support elements as an indicator). The goal of teamwork assessment in this situation is to recognize the effective collaborative process that led to the collective team's response. In this case the Prompt was directed at Intel, but Sustainment gave the good response, and in fact any role could have. As a result, when the Topic is marked as covered, and supplemented with information from the Lead / Instructor, this information is conveyed to a GIFT DKF at the team level.

This example illustrates the thinking behind the current design for how the GIFT DKFs are structured for this domain. The DKFs apply to the entire team as a whole. Although a Topic may have a notional association to the Intel function, any staff member may contribute to the discussion, and any positive or negative teamwork processes involved in the discussion are attributed to the entire team. If specific roles made key contributions, this can be preserved in the supplemental information conveyed with markers for the Topic, but at least from a representational perspective, the Topic is not strictly confined to the one role. The DKFs also essentially make use of Tasks that apply throughout the exercise, since there are no specific bounds on when a staff team may discuss a Topic. They may discuss a Topic before receiving a related Prompt from the Lead / Instructor, and they may also revisit a Topic after having moved on to other Topics or even after the original Topic has been marked as covered. One advantage to this approach is that it reflects a relatively simple structure for DKFs and team organization, rather than having large numbers of DKFs for individual, pair-wise, or n-wise subsets of roles.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

The design approach for interoperability with GIFT discussed in this paper is under development, so the intention is to document lessons learned from the development process in the future. Although there are complexities in fitting a training application for collaborative problem-solving to a structure designed for more enumerable, discrete, and scorable competencies, the tools in the GIFT framework provide an effective basis for building an approach that can be reused for similar domains.

A relevant research question in developing the integrated training approach is where the opportunities are for reusability across scenarios and potentially even training environments. This is an ongoing area of investigation from a couple of different perspectives. First, in terms of using the information embedded in markers about abstract team dimensions (team cognition, information exchange, etc.), a potential question is: if a team using this training applications demonstrates effective teamwork indicated by markers for certain dimensions, how likely is it that this represents findings that would hold true for the same team and same dimensions, but in other scenarios and even other collective tasks? There may be some dimensions that translate better than others; for example supporting behavior might be a recurring feature of a team, whereas information exchange or team cognition might be more closely related to individual knowledge.

Second, in terms of the teamwork markers tied to specific exercise Topics, there's a similar question: if the team performed well on that Topic for this scenario, would it be reasonable to expect them to perform well on the same Topic in a different scenario? This seems more likely to recur across scenarios, but it remains an area of investigation

This research focus was highlighted in a previous GIFT Sym paper (Jensen et al, 2022), which conveyed that the purpose of assessment and the review of teamwork markers generated in an exercise is initially limited to the scope of the exercise itself. However, using GIFT to enable the tracking of data over time, future development should expand the use case to situations where the same team composition engages in an exercise multiple times (or a series of exercises over time), where the goal is to see improvement among the team as a result of repeating the cycle and team review. As discussed earlier, this can become a complex team modeling problem when dealing with changes in team composition, but a baseline case can assume iterations with the same team composition.

One of the observations from instructors has been that front-end elements may also be an effective area for reuse. As the wargaming preparation training application may be more broadly considered a team familiarization exercise, the collection of user interface panels may also have value if implemented as reusable components to be readily adapted for other domains. This observation from instructors arose especially as a result of their experiences with challenges conducting team training events during the pandemic, where existing chat and teleconference tools became the norm but also fell short of instructional needs in certain ways. A common set of panels that can be composed into a browser-based training environment (map or image panel, chat / message panel, instructor tools, etc.) could be developed as a set of reusable front-end elements in the GIFT toolset, to support a potential cluster of related team training applications. Ultimately the aim is not solely to develop a single trainer for a specific application, but to derive lessons from this process for other similar future training needs.

Finally, another area for future work relates to the authoring of scenario content. Since a library of auto-complete messages plays a key role in the automated rules executed in the RAAFT training application, one of the future development goals is to grow the library over time by mining exercise data. When participants type free-text messages, and these are tagged in markers created by Lead / Instructor roles, the accumulated data set amounts to a source for semi-automated expansion of the library. Over time, this may reduce burdens on both instructors and authors.

REFERENCES

Grand, J.A., Pearce, M., Rench, T.A., Chao, G.T., Fernandez, R., & Kozlowski, S.W. (2013). Going DEEP: guidelines for building simulation-based team assessments. BMJ Qual Saf. 2013 May;22(5):436-48. doi: 10.1136/bmjqs-2012-000957. Epub 2013 Jan 25. PMID: 23355693.

Jensen, R., Presnell, B., DeFalco, J., and Goodwin, G. (2021). Designing a Distributed Trainer Using GIFT for Team Tutoring in Command Level Decision Making and Coordination. *In Proceedings of the 9th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym9).* 176-185.

Jensen, R., Finnigan, E., Teo, G., & Goodwin, G. (2022) Team Assessment Methods for Training Command Staff Decision Making. In *Proceedings of the 10th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym10)*. 157-167.

Salas, E., Shuffler, M. L., Thayer, A. L., Bedwell, W. L., & Lazzara, E. H. (2014). Understanding and improving teamwork in organizations: A scientifically based practical guide. *Human Resource Management*, 54(4), 599–622.

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

Teo, G., Jensen, R., Johnston, J., DeFalco, J., & Goodwin, G. (2021). Measures for Assessing Command Staff Team Performance in Wargaming Training. *Proceedings of the Interservice / Industry Training, Simulation, and Education Conference (I/ITSEC 2021).*

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 wargaming. Randy has a B.S. with honors from Stanford University.*

Evan Finnigan *is a software engineer at Stottler Henke Associates Inc., where his work includes developing intelligent tutoring software and building interactive web-based user interfaces. Previously, Evan was a student researcher in a human robot interaction lab where he built web-based interfaces to control an assistive robot. Evan has a B.S and M.S. from the University of California at Berkeley.*

Grace Teo, Ph.D. *is a Senior Research Psychologist at Quantum Improvements Consulting. Grace's research involves understanding and improving human performance under various conditions and in different contexts such as working with different technologies, and in teams. Other research interests include assessments, decision making processes and measures, vigilance performance, human-robot teaming, automation, and individual differences.*

Ms. Lisa Townsend *is a Psychologist who has been working at STTC (SED/CCDC SC) in Orlando, FL since August 2022. Prior to STTC, she spent 27 years as a Research Psychologist at the Naval Air Warfare Center Training Systems Division (NAWCTSD) in Orlando, FL. She has a Master of Science in Industrial/Organizational Psychology and a Bachelor of Arts in Psychology, both from the University of Central Florida (UCF). She has worked on many diverse teams including those within Research and Development, Technology Transfer, Instructional Systems Design, and Human Systems Integration, in efforts that have spanned across Services and platforms.*