The Generalized Intelligent Framework for Tutoring (GIFT)

Robert A. Sottilare, Keith W. Brawner, Benjamin S. Goldberg and Heather K. Holden
U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED)
SFC Paul Ray Smith Simulation & Training Technology Center (STTC)
Learning in Intelligent Tutoring Environments (LITE) Laboratory

INTRODUCTION

An emphasis on self-regulated learning in the military community (U.S. Army Training & Doctrine Command, 2011) has highlighted a need for point-of-need training in environments where human tutors are either unavailable or impractical. Computer-Based Tutoring Systems (CBTS) have been shown to be as effective as expert human tutors (VanLehn, 2011) in one-to-one tutoring in well-defined domains (e.g., mathematics or physics) and significantly better than traditional classroom training environments. CBTS have demonstrated significant promise, but fifty years of research have been unsuccessful in making CBTS ubiquitous in military training or the tool of choice in our educational system. Why?

The availability and use of CBTS have been constrained by their high development costs, their limited reuse, a lack of standards, and their inadequate adaptability to the needs of learners (Picard, 2006). Their application to military domains is further hampered by the complex and often ill-defined environments in which our military operates today. CBTS are often built as domain-specific, unique, one-of-a-kind, largely domain-dependent solutions focused on a single pedagogical strategy (e.g., model tracing or constraint-based approaches) when complex learning domains may require novel or hybrid approaches. The authors posit that a modular CBTS framework and standards could enhance reuse, support authoring and optimization of CBTS strategies for learning, and lower the cost and skillset needed for users to adopt CBTS solutions for military training and education. This paper considers the design and development of a modular CBTS framework called the Generalized Intelligent Framework for Tutoring (GIFT).

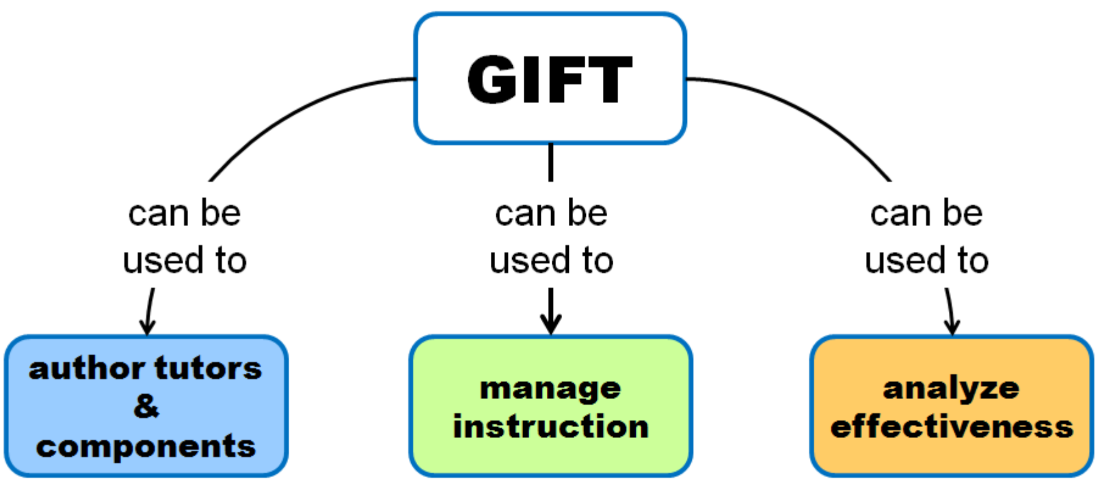

Figure 1: GIFT Ontology

GIFT has three primary functions (Figure 1). First, it is an authoring capability to develop new CBTS components, and whole tutoring systems. Second, GIFT is an instructional manager that integrates selected tutoring principles and strategies for use in CBTS. Finally, GIFT is an experimental testbed to analyze the effectiveness and impact of CBTS components, tools, and methods. GIFT is based on a learner-centric approach and a primary driver for tutoring research conducted by the Learning in Intelligent Tutoring Environments (LITE) Lab at Army Research Laboratory (ARL) is to improve linkages in the adaptive tutoring learning effect chain (Figure 2).

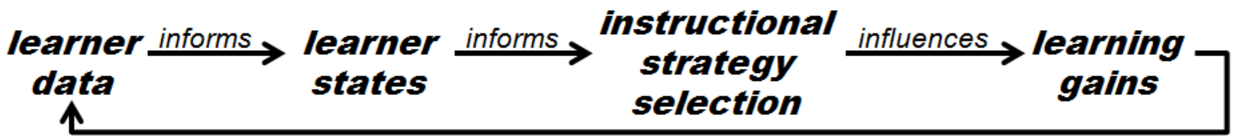

Figure 2: Adaptive Tutoring Learning Effect Chain (Sottilare, 2012)

A deeper understanding of the learner's behaviors, traits, and preferences (learner data) collected through performance, physiological and behavioral sensors and surveys will allow for more accurate evaluation of the learner's states (e.g., engagement level, confusion, frustration) which will result in a better and more persistent model of the learner. To enhance the adaptability of the CBTS, methods are needed to accurately classify learner states (e.g., cognitive, affective, psychomotor, social) and to select optimal instructional strategies given the learner's existing states. A more comprehensive learner model will allow the CBTS to adapt more appropriately to address the learner's needs by changing the instructional strategy (e.g., content, flow or feedback). An instructional strategy that is better aligned to the learners' needs is more likely to positively influence their learning gains.

GIFT 1.0 was released to U.S. Department of Defense organizations and their contractors (Distribution D) in May 2012. GIFT 2.0 is currently under testing and will be available to U.S. Government agencies and their contractors (Distribution C) in November 2012. The build schedule for GIFT has one release every six months over the next five years, and each release builds upon previous versions of the framework. Our goal is to realize an open-source (Distribution A) GIFT in the next year. For more information on GIFT, please visit GIFTtutoring.org.

GIFT DESIGN PRINCIPLES

The methodology for the development of a modular, computer-based tutoring framework for military training considered major design goals, anticipated uses and applications. The design process also considered enhancing one-to-one (individual) and one-to-many (collective or team) training experiences beyond the state of practice for CBTS today. A significant focus of the GIFT design was to concentrate domain-dependent elements in the domain module. This was done to allow large scale reuse of the remaining GIFT modules across different training domains and thereby reduce the development costs for CBTS.

One design principle adopted in GIFT is that each module should be capable of gathering information from other modules according to the design specification. Designing to this principle resulted in standard message sets and message transmission rules (i.e., request-driven, event-driven or periodic transmissions). For instance, the pedagogical module is capable of receiving information from the learner module to develop courses of action for future instructional content to be displayed, to manage flow and challenge level, and to select appropriate feedback. Changes to the learner's state (e.g., engagement, motivation or affect) trigger messages to the pedagogical module which then recommends general courses of action (e.g., ask a question or prompt the learner for more information) to the domain module which provides a domain-specific intervention (e.g., what is the next step?).

Another design principle adopted within GIFT is the separation of content from the executable code (Patil & Abraham, 2010). Data and data structures are placed within models and libraries, while software processes are programmed into interoperable modules. Additional design considerations were given to efficiency and effectiveness goals (e.g., accelerated learning and enhanced retention) for training given the time available for military training and renewed emphasis on self-regulated learning. An outgrowth of this emphasis on efficiency and effectiveness led the authors to seek external guidance. In 2012, ARL with the University of Memphis developed advisory boards of senior tutoring system researchers from academia and government to influence the GIFT design goals moving forward. An advisory board for learner modeling was completed in September 2012, and future boards are planned for instructional strategy design, tool authoring and expert modeling, learning effect evaluations, and domain modeling.

Design goals and anticipated uses

As noted previously, the three primary functions of GIFT are in the areas of authoring, instruction, and analysis of effect. Discussed below are the purpose, associated design goals and anticipated uses for each of these functional areas or "constructs."

GIFT Authoring Construct

The purpose of the GIFT authoring construct is to provide technology (tools and methods) to make it affordable and easier to build CBTS and CBTS components. Toward this end, a set of XML configuration tools continues to be developed to allow for data-driven changes to the design and implementation of GIFT-generated CBTS. The design goals for the GIFT authoring construct have been adapted from Murray (1999, 2003) and Sottilare & Gilbert (2011). The GIFT authoring design goals are:

- Decrease the effort (time, cost, and/or other resources) for authoring and analyzing CBTS by automating authoring processes, developing authoring tools and methods, and developing standards to promote reuse
- Decrease the skill threshold by tailoring tools for specific disciplines (e.g., instructional designers, training developers, and trainers) to author, analyze and employ CBTS technologies
- Provide tools to aid designers/authors/trainers/researchers in organizing their knowledge
- Support (structure, recommend, or enforce) good design principles in pedagogy through user interfaces and other interactions
- Enable rapid prototyping of CBTS to allow for rapid design/evaluation cycles of prototype capabilities
- Employ standards to support rapid integration of external training/tutoring environments (e.g., simulators, serious games, slide presentations, transmedia narratives, and other interactive multimedia)
- Develop/exploit common tools and user interfaces to adapt CBTS design through data-driven means
- Promote reuse through domain-independent modules and data structures
- Leverage open source solutions to reduce CBTS development and sustainment costs
- Develop interfaces/gateways to widely used commercial and academics tools (e.g., games, sensors, toolkits, virtual humans)

As a user-centric architecture, anticipated uses for GIFT authoring tools are driven largely by the anticipated users which include learners, domain experts, instructional system designers, training & tutoring system developers, trainers & teachers, and researchers. In addition to user models and graphical user interfaces, GIFT authoring tools include domain-specific knowledge configuration tools, instructional strategy development tools and a compiler to generate executable CBTS from GIFT components in a variety of formats (e.g., PC, Android, and iPad).

Within GIFT, domain-specific knowledge configuration tools include authoring of new knowledge elements or the reuse of existing (stored) knowledge elements. Domain knowledge elements include learning objectives, media, task descriptions, task conditions, standards and measures of success, common misconceptions, feedback library, and a question library which are informed by instructional system design principles, and in turn inform concept maps for lessons and whole courses. The task descriptions, task conditions, standards and measures of success, and common misconceptions may be informed by an expert or ideal learner model derived through a task analysis of the behaviors of a highly skilled user. ARL is investigating techniques to automate this expert model development process to reduce the time and cost of developing CBTS. In addition to feedback and questions, additional tools are anticipated to author explanations, summaries, examples, analogies, hints, and prompts in support of GIFT's instructional management construct.

GIFT Instructional Management Construct

The purpose of the GIFT instructional management construct is to integrate pedagogical best practices in GIFT-generated CBTS. The modularity of GIFT will also allow GIFT users to extract pedagogical models for use in tutoring/training systems that are not GIFT-generated. GIFT users may also integrate pedagogical models, instructional strategies, or instructional tactics from other tutoring systems into GIFT. The design goals for the GIFT instructional management construct are:

- Support CBTS instruction for individuals and small teams in local and geographically distributed training environments (e.g., mobile training), and in both well-defined and ill-defined learning domains
- Provide for comprehensive learner models that incorporate learner states, traits, demographics, and historical data (e.g., performance) to inform CBTS decisions to adapt training/tutoring
- Support low-cost, unobtrusive (passive) methods to sense learner behaviors and physiological measures and use this data along with instructional context to inform models to classify (in near real-time) the learner's states (e.g., cognitive and affective)
- Support both macro-adaptive strategies (adaptation based on pre-training learner traits) and micro-adaptive instructional strategies and tactics (adaptation based learner states and state changes during training)
- Support the consideration of individual differences where they have empirically been documented to be significant influencers of learning outcomes (e.g., knowledge or skill acquisition, retention, and performance)
- Support adaptation (e.g., pace, flow, and challenge level) of the instruction based the domain and learning class (e.g., cognitive learning, affective learning, psychomotor learning, social learning)
- Model appropriate instructional strategies and tactics of expert human tutors to develop a comprehensive pedagogical model

To support the development of optimized instructional strategies and tactics, GIFT is heavily grounded in learning theory, tutoring theory and motivational theory. Learning theory applied in GIFT includes cognitive learning (Anderson & Krathwohl, 2001), affective learning (Krathwohl, Bloom, and Masia, 1964; Goleman, 1995), psychomotor learning (Simpson, 1972), and social learning (Sottilare, Holden, Brawner, and Goldberg, 2011; Soller, 2001). Aligning with our goal to model expert human tutors, GIFT considers the INSPIRE model of tutoring success (Lepper, Drake, and O'Donnell-Johnson, 1997) and the tutoring process defined by Person, Kreuz, Zwaan, and Graesser (1995) in the development of GIFT instructional strategies and tactics.

INSPIRE is an acronym that highlights the seven critical characteristics of successful tutors: Intelligent, Nurturant, Socratic, Progressive, Indirect, Reflective and Encouraging. Person, Kreuz, Zwaan, and Graesser's tutoring process includes a tutor-learner interchange where the tutor asks a question, the learner answers the question, the tutor gives feedback on the answer, then the tutor and learner collaboratively improve the quality of (or embellish) the answer. Finally, the tutor evaluates the learner's understanding of the answer.

As a learner-centric architecture, anticipated uses for GIFT instructional management capabilities include both automated instruction and blended instruction where human tutors/teachers/trainers use GIFT to support their curriculum objectives. If its design goals are realized, it is anticipated that GIFT will be widely used beyond military training contexts as GIFT users expand the number and type of learning domains and resulting CBTS generated using GIFT.

GIFT Analysis Construct

The purpose of the GIFT analysis construct is to allow CBTS researchers to experimentally assess and evaluate CBTS technologies (CBTS components, tools, and methods). The design goals for the GIFT analysis construct are:

- Support the conduct of formative assessments to improve learning
- Support summative evaluations to gauge the effect of technologies on learning
- Support assessment of CBTS processes to understand how learning is progressing throughout the tutoring process
- Support evaluation of resulting learning versus stated learning objectives
- Provide diagnostics to identify areas for improvement within CBTS processes
- Support the ability to comparatively evaluate CBTS technologies against traditional tutoring or classroom teaching methods
- Develop a testbed methodology to support assessments and evaluations (Figure 3)

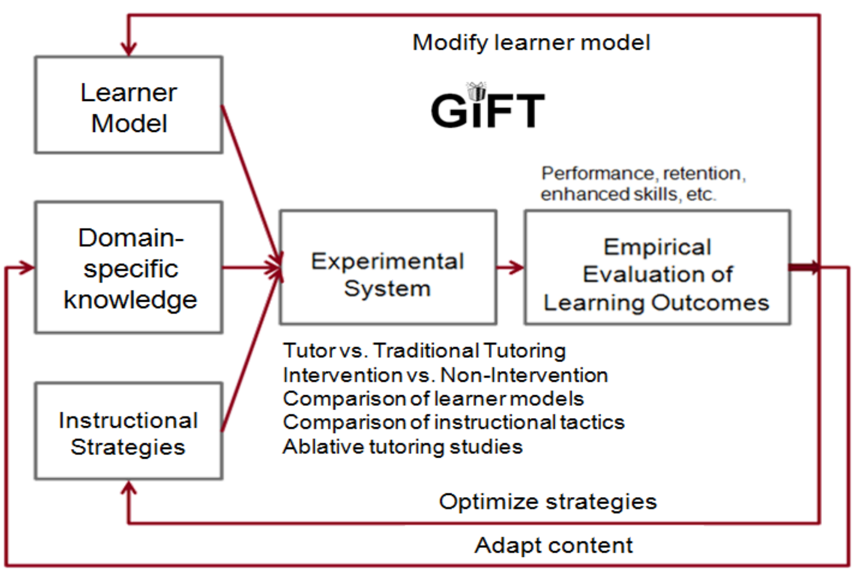

Figure 3: GIFT Analysis Testbed Methodology

Figure 3 illustrates an analysis testbed methodology being implemented in GIFT. This methodology was derived from Hanks, Pollack, and Cohen (1993) to allow manipulation of the learner model, instructional strategies, and domain-specific knowledge within GIFT, and support analysis of artificially-intelligent agents that influence the adaptive tutoring learning effect chain. In developing their testbed methodology, Hanks et al. reviewed four testbed implementations (Tileworld, the Michigan Intelligent Coordination Experiment [MICE], the Phoenix testbed and Truckworld) for evaluating the performance of artificially-intelligent agents. Although agents have changed substantially in complexity during the past 20-25 years, the methods to evaluate their performance have remained markedly similar.

The authors designed the GIFT analysis testbed upon Cohen's assertion (Hanks et al., 1993) that testbeds have three critical roles related to the three phases of research. During the exploratory phase, agent behaviors need to be observed and classified in broad categories. This can be performed in an experimental environment. During the confirmatory phase, the testbed is needed to allow more strict characterizations of agent behavior to test specific hypotheses and compare methodologies. Finally, in order to generalize results, measurement and replication of conditions must be possible. Similarly, the GIFT analysis methodology (Figure 3) enables the comparison/contrast of CBTS elements and assessment of their effect on learning outcomes (e.g., knowledge acquisition, skill acquisition and retention).

GIFT FUNCTIONAL ELEMENTS

The GIFT functional elements include the components, modules, models, messages, databases, libraries and interfaces that support the authoring, instructional and analysis processes within GIFT. The major GIFT functional elements and interactions are shown in the GIFT Functional Block Diagram (Figure 4) and are discussed below.

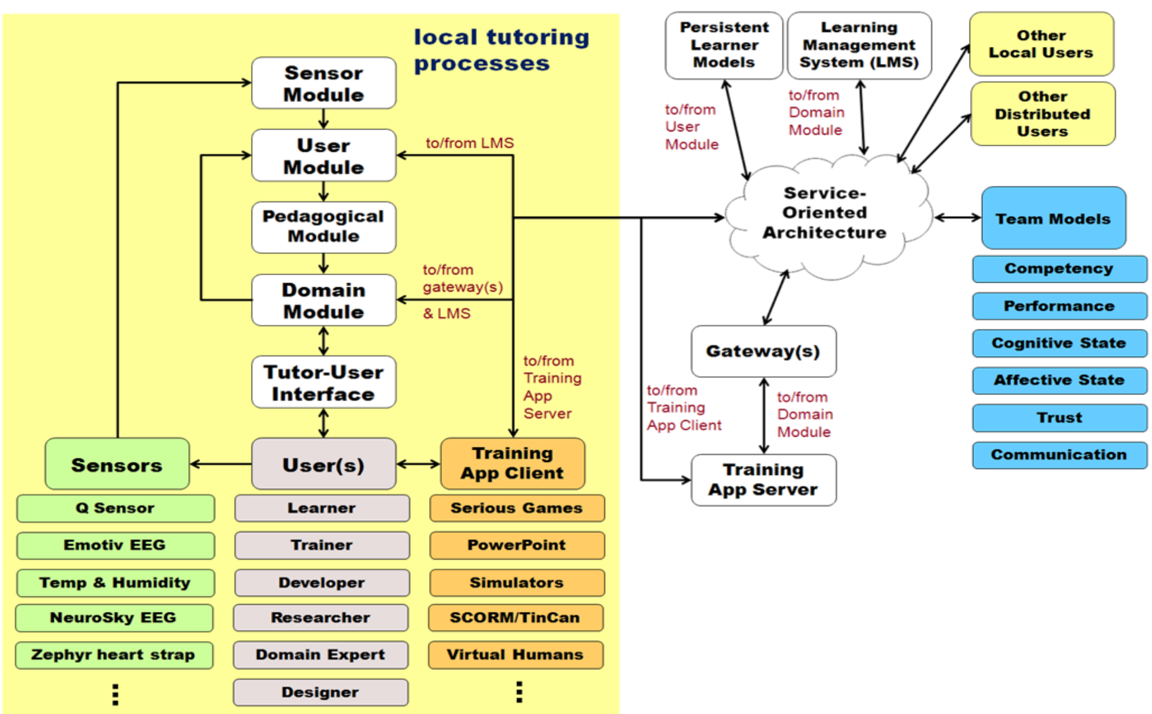

Figure 4: GIFT Functional Block Diagram

GIFT Service-Oriented Architecture

GIFT incorporates a Service-Oriented Architecture (SOA) to support distributed team training and mobile learning contexts which extend tutoring beyond the traditional one-to-one contexts. Tutoring processes (e.g., sensing and learner state assessment, and content presentation) must be able to support tutoring that is physically isolated from the learner. For example, the CBTS for the mobile learner must support local processes for the sensors used to gather the learner's data, but the learner module that assesses the learner's state may not be local due to the computation limitations of the mobile device.

For each module to interact seamlessly and adapt to new information, a JAVA-based framework has been constructed to facilitate the communication of GIFT modules. The SOA allows GIFT to be expanded by adding new modules and standard messages. GIFT is also able to run on physically separate computers/mobile devices and to be accessed by multiple learners simultaneously to support concurrent training sessions or distributed team training. JBoss, a JAVA-based application server, has been implemented in GIFT to publish and receive information not initially expected in the design, provided that the information recipient is compatible. Messages are JavaScript Object Notation (JSON) encoded.

The GIFT design allows each module to run as a separate process to be independent of the computer programming language. The tradeoff of this design decision is that each module must have access to shared routines. These routines perform the function of communicating within the framework, and include functions such as sending, receiving, encoding, and parsing messages. The shared functions are part of an initial library of shared code which is currently written in JAVA and C++. For module developers to be able to program in other languages this library must be written into other programming languages.

Sensors and the GIFT Sensor Module (domain-independent)

As shown in Figure 5, the sensor module contains a suite of specific sensor interfaces for behavioral and physiological sensors. It accepts raw sensor data and adapts this data into a usable form by a sensor processing module which filters, segments or extracts features in the data. The sensor processing module then transfers the processed data to Learner Module via the SOA. The fact that each sensor used in the architecture is different indicates that there must be an interface and an associated sensor processing module for each sensor used. Since there is an associated processing cost with the implementation of each sensor, it is critical that minimum sets of sensors be identified to determine each cognitive and affective state within different training platforms (e.g., computer-based training, mobile learning).

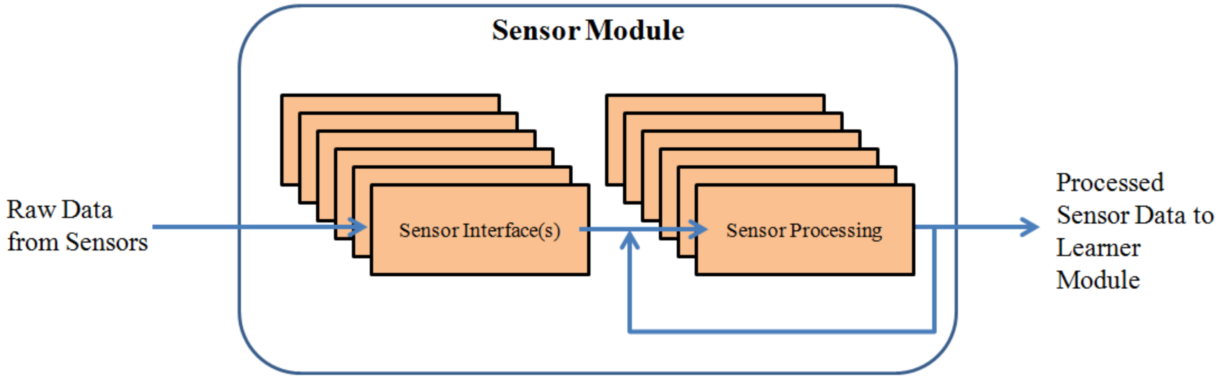

Figure 5: GIFT Domain-Independent Sensor Module

GIFT User Module, User Models and Interfaces

The user module is primarily driven by the user model and interface selected. If the user is a learner, the user module becomes a learner module and supports the assessment of the learner's states (e.g., cognitive, affective or physical) based on historical data (e.g., performance, traits, preferences), survey data, and sensor data. The primary function of the learner module is to determine the learner's states, but a secondary function is to track relevant learner trait data (e.g., personality preferences and learning style) and lesson data (e.g., lesson start/completion, lesson scores or performance) for the instructional process. Additional user models and graphical user interfaces are being developed to support instructional designers, subject matter experts, researchers, developers and trainers/teachers/tutors, but the learner model remains at the center of providing an adaptive tutoring capability.

The learner module uses pre-processed behavioral and physiological data from the sensor module, and a performance assessment (e.g., failing to meet, meeting or exceeding expectations) along with demographic, self-reported and observed data to classify the learner's cognitive, affective and competency states. This learner state data is used by the pedagogical module to determine which content or instructional strategy (e.g., direction, questions, feedback) to present to the learner next.

The learner module assesses the current state of the learner and predicts future states. For example, this module predicts future performance based on the learner's current state by evaluating the likelihood of switching states to higher or lower levels of performance. As shown in Figure 6, the learner module uses sensor input from the sensor module and performance input from the pedagogical module along with historical data (e.g., personality assessments, domain experience) to predict learner state using semi-supervised machine learning techniques such as clustering, hidden Markov models and likelihood estimation. In semi-supervised machine learning both labeled and unlabeled data is used for training the classifier which is used to determine performance, cognition or affect.

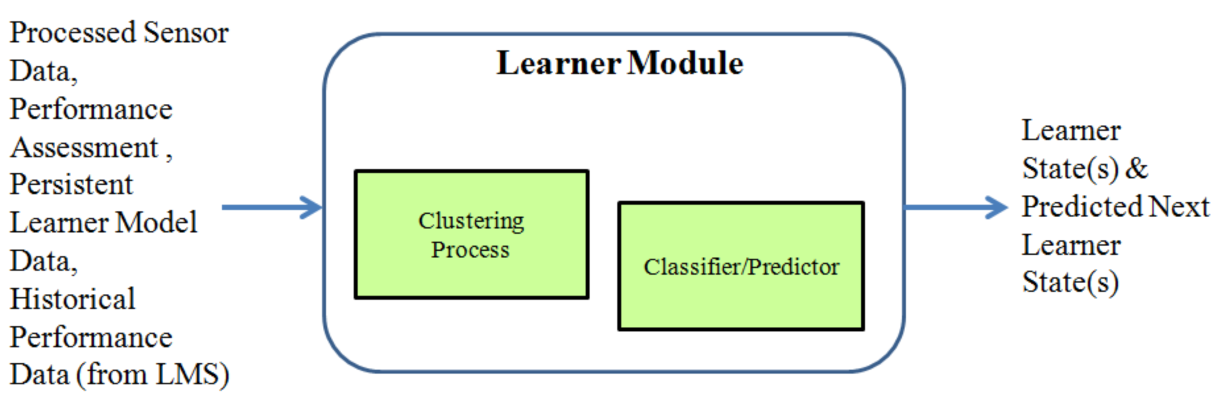

Figure 6: GIFT Domain-Independent Learner Module

There are fundamental technical challenges and different approaches for developing a learner module. For instance, while one learner module author may select to use a system based on previously established threshold values of sensory input, another learner module author may decide to use an unsupervised clustering method of determining states. GIFT allows pedagogical strategy decisions to be made as a function of either one of these approaches. Given the modular nature of GIFT and its testbed capabilities, individual learner module approaches can be compared empirically against each other fairly and objectively.

GIFT Pedagogical Module (domain-independent)

As shown in Figure 7, the pedagogical module queries state data from the learner module (learner state) and the domain module (performance assessment and associated feedback). It uses learner state and performance to determine the content, order and flow of instruction. Visual and audio stimuli are fed to the user-tutor interface based on recommendations by the pedagogical module. The question that the pedagogical module is tasked to answer is: "given a learner is in the following state, what is the recommended course of action?" Traditional machine learning methods such as Bayesian networks or decision trees can be used in making the decision regarding pedagogical strategy. Pedagogical strategies are in place to make decisions about what to do next when multiple options are available (Dabbagh, 2005). They can manipulate elements within a training scenario; provide hints or feedback; and change the pace and difficulty of interaction (Wulfeck, 2009).

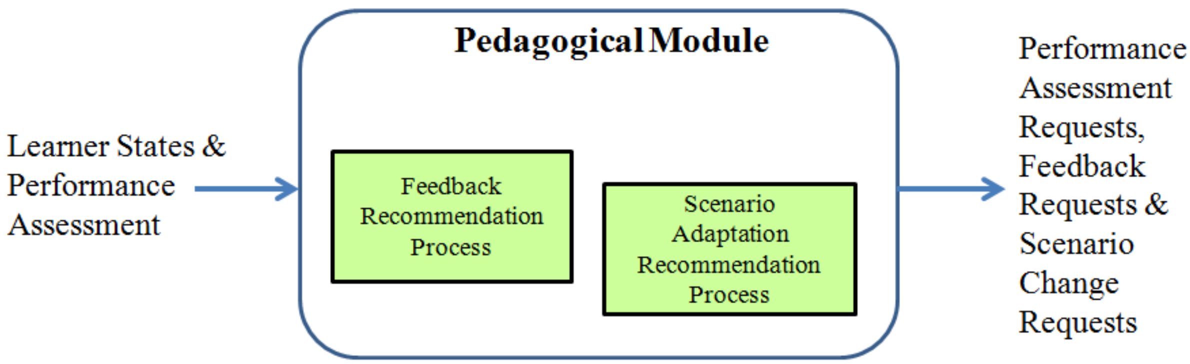

Figure 7: GIFT Domain-Independent Pedagogical Module

GIFT provides a medium for evaluating the effectiveness of multiple pedagogical strategies on performance across various domains. The architecture also supports experimentation for comparing data sets with a goal of identifying variables that trigger optimal strategies based on performance and diagnosed states. The objective is to determine what elements affect training and to determine what characteristics of an individual should be used to select a mediation or support strategy based on that affect.

Once again, the development of a pedagogical strategy module is a fundamental technical challenge. GIFT allows for the selection of instructional strategy based on multi-dimensional learner state. Additionally, GIFT makes the comparison of the performance of different pedagogical strategies in different domains and contexts possible. Initially, it is expected that learner state will be utilized to classify learners into different levels of mastery, and that a decision tree or Markov process will be utilized to make optimal instructional strategy recommendations.

GIFT Domain Module (domain-dependent)

The purpose of the domain module is to define and structure domain knowledge (e.g., instructional content, domain-relevant tasks/conditions/standards, problem sets, common questions and common misconceptions). Several standards exist that represent domain knowledge: the Institute of Electrical and Electronics Engineers, Inc. (IEEE) Learning Object Metadata (LOM); Sharable Content Object Reference Model (SCORM) and Instrument and Measurement Society (IMS) content packaging (Boneu, 2011). GIFT is being designed to interface with any or all of these standards in order to leverage domain-specific content that is compatible with these standards.

As shown in Figure 8, the actions of the domain module are driven by the pedagogical module which recommends generalized strategies (e.g., ask a question, prompt the learner for more information, or review basic concepts) which the domain module converts into specific instructional tactics (e.g. ask a question on concept B, prompt the learner for more information on concept B, or review basic concepts for building clearing tasks). The decisions on which content to present, the order and pace of presentation, and the type of feedback to provide (e.g., supportive, directive hints or questions) is made by the domain module. The domain module also assesses learner actions and progress over a lesson, course or area of study. This is generally accomplished by comparing the correctness of learner responses, and comparing learner performance to expert performance or other standards. By maintaining the learner assessment within the domain module and then only providing the result of the learner assessment to the pedagogical module, the domain-independence of the pedagogical module and other functions within GIFT are maintained.

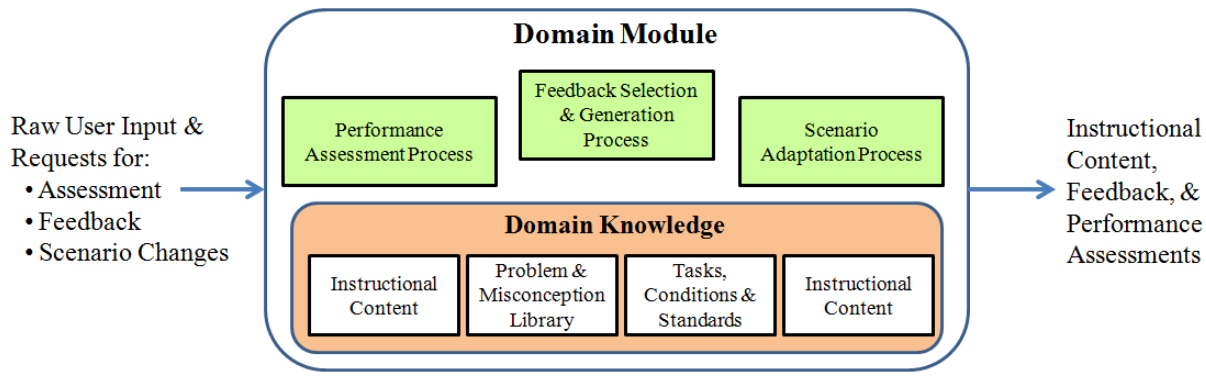

Figure 8: GIFT Domain Module

The Domain Module captures data that is coming out of the training environment (e.g., game or simulator), and assesses performance against key concepts. These assessments are sent out to the Learner Module and compared to previous assessments to determine learning trends. Each of the assessments is labeled as above standard, at standard, below standard, or unknown so that all of the other modules can treat it as if it was any other concept. The concepts that are assessed are parts of tasks and tasks are in turn part of lessons. All task assessments address what is done, the conditions under which the task is done, and the standards or measures under which the task performance is assessed.

Training Application Servers, Clients and GIFT Gateways

To leverage external training environments, GIFT has developed gateways to interface multi-media training environments to the SOA using application servers and clients for serious games, simulators, distributed learning environments (e.g., SCORM or Experience API), and virtual human capabilities. To support a higher level of interaction and an adaptive tutoring capability, GIFT has also developed a gateway to integrate presentation-based (e.g., PowerPoint) courses/lessons.

GIFT Team Models

When learners train as part of a group, they can encourage each other to ask questions, explain/justify their opinions and reasoning, and actively reflect upon their knowledge. Research has shown such situations to increase group performance and individual learning outcomes (especially motivation and engagement - Tchounikine, Rummel, and McLaren, 2010). However, these benefits can only be achieved in well-functioning, actively learning teams (Jarboe, 1996; Soller, 2001). While some teams may have successful interaction and communication naturally, others may be incapable of developing a balance of participation, leadership, understanding, and encouragement (Soller, 2001). This dysfunction can rapidly degrade group and individual performance, motivation, and engagement.

In order for a CBTS to successfully support team training/tutoring, three primary factors should be understood and addressed in its design: (a) the collaborative learning interactions and communications; (b) the complexity of training tasks; and (c) the physical distribution of the team. GIFT has been designed to support collaborative interaction/communication of local and distributed teams across a wide variety of training tasks. To this end, GIFT includes team models to assess team competency, performance, cognitive state, affective state, trust, and communication.

Learning Management Systems & GIFT Persistent User Models

To support a more long term (persistent) view of learners and teams, GIFT has plans to integrate with a set of Learning Management Systems including commercial systems (e.g., Moodle) to capture long term performance. GIFT is also developing a persistent model to maintain a long term view of the learner's states, traits, demographics, preferences, and historical data (e.g., survey results, performance, competencies).

REFERENCES:

Anderson, L. W., and Krathwohl, D. R. (Eds.). (2001). *A taxonomy for learning, teaching and assessing: A revision of Bloom's Taxonomy of Educational Objectives: Complete edition*, New York : Longman.

Boneu, J. M. (2011). Survey on Learning Content Management Systems. In N.F. Ferrer, & J.M. Alfonso (Eds.), *Content Management for e-Learning* (pp. 113-130). New York: Springer.

Dabbagh, N. (2005). Pedagogical Models for E-Learning: A Theory-Based Design Framework. *International Journal of Technology in Teaching and Learning*, 1(1), 25-44.

Goleman, D. (1995). *Emotional Intelligence*. New York: Bantam Books.

Hanks, S., Pollack, M.E., & Cohen, P.R. (1993). Benchmarks, Test Beds, Controlled Experimentation, and the Design of Agent Architectures. *AI Magazine*, 14 (4), 17-42.

Jarboe, S. (1996). Procedures for enhancing group decision making. In B. Hirokawa & M. Poole (Eds.), *Communication and Group Decision Making* (pp. 345-383). Thousand Oaks, CA: Sage Publications.

Krathwohl, D.R., Bloom, B.S., & Masia, B.B. (1964). *Taxonomy of Educational Objectives: Handbook II: Affective Domain*. New York: David McKay Co.

Lepper, M. R., Drake, M., & O'Donnell-Johnson, T. M. (1997). Scaffolding techniques of expert human tutors. In K. Hogan & M. Pressley (Eds), Scaffolding learner learning: Instructional approaches and issues (pp. 108-144). New York: Brookline Books.

Murray, T. (1999). Authoring intelligent tutoring systems: An analysis of the state of the art. *International Journal of Artificial Intelligence in Education*, 10(1):98–129.

Murray, T. (2003). An Overview of Intelligent Tutoring System Authoring Tools: Updated analysis of the state of the art. In *Authoring tools for advanced technology learning environments*. pp. 491-545.

Patil, A. S., & Abraham, A. (2010). Intelligent and Interactive Web-Based Tutoring System in Engineering Education: Reviews, Perspectives and Development. In F. Xhafa, S. Caballe, A. Abraham, T. Daradoumis, & A. Juan Perez (Eds.), *Computational Intelligence for Technology Enhanced Learning. Studies in Computational Intelligence* (Vol 273, pp. 79-97). Berlin: Springer-Verlag.

Person, N. K., Kreuz, R. J., Zwaan, R. A., & Graesser, A. C. (1995). Pragmatics and pedagogy: Conversational rules and politeness strategies may inhibit effective tutoring. *Cognition and Instruction*, 13(2), 161–188.

Picard, R. (2006). Building an Affective Learning Companion. Keynote address at the 8th International Conference on Intelligent Tutoring Systems, Jhongli, Taiwan. Retrieved from http://www.its2006.org/ITS_keynote/ITS2006_01.pdf

Simpson, E. (1972). The classification of educational objectives in the psychomotor domain: *The psychomotor domain*. Vol. 3. Washington, DC: Gryphon House.

Soller, A. (2001). Supporting social interaction in an intelligent collaborative learning system. International Journal of Artificial Intelligence in Education, 12(1), 40-62.

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

Sottilare, R., Holden, H., Brawner, K., & Goldberg, B. (2011). Challenges and Emerging Concepts in the Development of Adaptive, Computer-based Tutoring Systems for Team Training. *Interservice/Industry Training Systems & Education Conference*, Orlando, Florida, December 2011.

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

Tchounikine, P., Rummel, N., & McLaren, B.M. (2010). “Computer Supported Collaborative Learning and Intelligent Tutoring Systems.” In R. Nkambo, J.Bourdeau, & R. Mizoguchi (Eds.) *Advances in Intelligent Tutoring Systems*. Chapter 22, 447-463, Berlin: Springer.

U.S. Army Training & Doctrine Command (2011). The United States Army Learning Concept for 2015. Fort Monroe, VA.

Wulfeck, W. H. (2009). *Adapting Instruction*. In Proceedings of the 5th International Conference on Foundations of Augmented Cognition (FAC'09): Held as Part of HCI International 2009, San Diego, CA.

VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems and other tutoring systems. *Educational Psychologist*, 46 (4) 197-221.