Exploring Methods to Promote Interoperability in Adaptive Instructional Systems

Robert Sottilare[(⊠)] [iD]

Soar Technology, Inc., Orlando, FL 32817, USA
bob.sottilare@soartech.com

Abstract. This paper explores design principles and methods to promote interoperability and reuse in a class of instructional technologies called adaptive instructional systems (AISs) which include intelligent tutoring systems (ITSs), intelligent mentors or recommender systems, and intelligent instructional media. AISs are *artificially-intelligent, computer-based systems that guide learning experiences by tailoring instruction and recommendations based on the goals, needs, and preferences of each individual learner or team in the context of domain learning objectives*. IEEE is exploring standards and best practices for AIS modeling, interoperability, and evaluation under its Project 2247 and affiliated working group. This paper was composed to document recommendations related to interoperability standards for AISs. The after school market for AISs is large and on the rise in China, the US, and in Europe. The desired level of interoperability for AISs is at the lowest possible level to allow component reuse without impinging upon the intellectual property of the vendor.

Keywords: Adaptive instructional system (AIS) ·
Intelligent tutoring systems (ITSs) · Interoperability · Reuse · Standards

1 Introduction

During the last year, an IEEE working group chartered through Project 2247 has taken on the task of developing standards and best practices for adaptive AISs. This IEEE working group will be debating what is and is not an AIS, and what standards and best practices will evolve from the marketplace. To date, the group has identified three potential areas for standardization: (1) a conceptual model for AISs, (2) interoperability standards for AISs, and (3) evaluation best practices for AISs. This paper explores design principles and methods to enhance interoperability and reuse for adaptive instructional systems (AISs) that are defined as: *artificially-intelligent, computer-based systems that guide learning experiences by tailoring instruction and recommendations based on the goals, needs, and preferences of each individual learner or team in the context of domain learning objectives* [1].

Adaptive instructional systems (AISs) come in many forms and this makes standardization challenging. The most common form of AIS is the intelligent tutoring system (ITS) which is a computer-based system that automatically provides real-time,

tailored, and relevant feedback and instruction to learners [2, 3]. Other forms of AISs include intelligent mentors (recommender systems) which promote the social relationship between learners and intelligent agents [4] and web-based intelligent media used for instruction.

Interoperability is defined as "the ability of two or more software components to cooperate despite differences in language, interface, and execution platform. It is a scalable form of reusability…" [5]. When examining interoperability as a design goal, we are attempting to design interfaces that are defined to a degree that allows information to be exchanged with and understood by other systems or system components [1].

Effective AISs allow information to flow between components and to be used by the components to enable transition from less optimal learning states to new positive learning states. This is accomplished by changes to the level/type of support, feedback, and difficulty of the content. AIS decisions may be triggered by trends in learner data (e.g., behavioral data), traits (e.g., personality traits) or states (e.g., behavioral or physiological states, emotions, learning, and performance). Data and traits are shared by the components and may be used to derive current states or project future states. Information (all data, traits, or states) are used to guide AIS decisions on the selection of domain-independent strategies (e.g., ask a question, provide support, prompt learner for more information, and request a reflective dialogue). Based on the strategy selection, tactics or specific domain-dependent actions are selected by the AIS. This sequence of interactions is conducted with the goal of optimizing tutoring decisions and ultimately learning. For AISs, this set of interactions has been described in the learning effect model (LEM) for both individual learners and teams of learners [6–10]. We are suggesting that the interactions between common AIS components, which are often arranged as encoded messages, might be our best near term opportunity for standardizing AISs. Interactions within and external to the AIS involve its four common components: a domain model, an individual learner or team model, an instructional model, and an interface model [11], but will all AISs have these common components? Will these components truly function the same to the degree that we can drop a domain model from one AIS into another AIS? These questions remain open until we have an approved conceptual model for AISs. Development of that formal conceptual model is still about a year away, but we can make some credible assumptions about these common components and how they function.

1.1   Interoperability in Domain Models

The *domain model* describes the set of skills, knowledge, strategies (plans for action), and tactics (actions executed by the ITS) for the topic/domain being instructed [11]. In the case of model tracing tutors, the domain model contains the expert knowledge in the form of an ideal learner model, and also includes the bugs, mal-rules, and misconceptions that students periodically exhibit. In the case of constraint-based tutors, the domain model includes constraints that have three data elements [12]:

- Relevance Condition – describes when the constraint is applicable
- Satisfaction Condition – specifies assessments to be applied to ascertain the correctness of the solution

- Feedback Message – communicates with the learner to advise them that their solution is incorrect and why it is incorrect, and provides reminds the learner of corresponding declarative knowledge

ITSs use the context or constraints found in domain models along with learner states to select optimal instructional strategies. The key attribute in AISs is intelligence or the ability to observe and adapt. Noting that this is critical to being an AIS, we advocate that domain models should be included as common components of all AISs. It should also be emphasized that the intellectual property in AISs resides primarily in the domain model in the form of content, methods of assessment, and instructional strategies and tactics. This makes interoperability much more difficult than in other AIS components, but even so, there could be opportunities for interoperable AIS domain models at the common component level by standardizing the format of the data that passes between domain models and other common models (i.e., learner, instructional, and interface models).

1.2   Interoperability in Individual Learner and Team Models

Individual learner models consist of the cognitive, affective, motivational, and other psychological states that evolve during instruction and moderate (enhance or minimize) learning. Often, the data required to classify or predict learner states is acquired from sensors, learner input (e.g., self-reported data), learner assessments (e.g., tests or tasks used to define a level of learner proficiency within a domain of instruction), or historical databases (e.g., long-term learner model, learning record store or learning management system). There are opportunities to standardize the format of this data to allow it to be used by AISs and we shall revisit this topic later in this paper.

Since learner performance is primarily tracked in the domain model, the learner model is often viewed as an overlay (subset) of the domain model, which changes over the course of tutoring. For example, "knowledge tracing" tracks the learner's progress from problem to problem and builds a profile of strengths and weaknesses relative to the domain model [13].

Team models are not specifically called out as ITS common components, but the focus on the dynamics and functions of teams in collaborative learning and collaborative problem solving has placed new emphasis on the design of ITSs for team use [11, 14–16]. The team model, sometimes referred to as a collective model, must be able to track progress toward collective task learning objectives for either training or collaborative learning goals. To support team development and enhance collaboration skills, team models are significantly more complex than individual learner models which are primarily focused on the assessment of tasks. Team models also assess teamwork which is the "coordination, cooperation, and communication among individuals to achieve a shared goal" [17].

ITSs use models of individual learners or teams to assess their progress toward defined learning objectives and drive instructional decisions to optimize learning. While some learner attributes are relatively static, most are of the attributes in ITS learner models are unique to the task or domain of instruction. Again, this poses a significant challenge to standardizing learner or team models, but standardized data formats could

support interoperability by allowing the exchange of information between learner models and other components (i.e., domain, instructional or interface model).

1.3   Interoperability in Instructional Models

The instructional model (also known as the pedagogical model or the tutor model) takes conditions from the domain model and states from the learner models as input and makes recommendations in the form of tutoring strategies (plans for future AIS actions), next steps, and tactics (actions on what the tutor should do next in the exchange with the learner). In mixed-initiative systems, the learners may also take actions, ask questions, or request help [18, 19], but the AIS should to be ready in real-time to decide "what to do next" at any point and this is determined by an instructional model that captures pedagogical theories in the form of recommended instructional practices.

The instructional model has a duality to it that encompasses generalizable theories of learning and instructional practices, and principles and measures of assessment that are specifically linked to particular domains of instruction. As we begin to think about the interoperability of AISs, we should consider open and proprietary solutions. Open solutions might be generalizable across domains or specific to a domain of instruction. Proprietary solutions may also be generalizable and specific. Either way, we must seek opportunities to standardize in areas that do not violate intellectual capital, and do not limit the movement of effective solutions into the marketplace. One way to accomplish this might be to treat these solutions as black boxes and standardize the information flowing into and out of the black boxes rather than attempt to dictate how the processes within are constructed. To this end, next we explore the fourth common element in AIS architectures, the interface model.

1.4   Interoperability in Interface Models

The *interface model* interprets the learner's contributions through various input media (speech, typing, clicking) and produces output in different media (text, diagrams, animations, agents). In addition to the conventional human-computer interface features, some recent systems have incorporated natural language interaction [20–22], speech recognition [23, 24], and the sensing of learner emotions [25–27].

The interface model also governs how learners interact within the AIS. It should be noted that part of the interface model is target directly at providing the instructional model with information needed to tailor the training and part of the interface model is targeted at the learner's interaction with the environment where the environment might be defined as an external simulation, some media, or a set of mathematical problems for the learner to solve. The interface captures behaviors related to interaction with the instructional content and provides this information to the instructional model within the AIS so it can update learner states and environmental conditions (e.g., context – where the learner is in the map of items to be learned) and then the AIS can make decisions with the goal of optimizing learning outcomes.

When we think about interface models, we naturally think about the flow of information through the AIS architecture: learner data captured by various sources as

discussed earlier in this paper, learner states derived from learner data, instructional options and recommendations for next steps (strategies and tactics). The data that flows from one model to another usually takes on the form of messages or blocks of data with specific formatting, but the message types and formats vary from one AIS architecture to another and this is problematic when our goal is interoperability and reuse of tools and methods across AISs.

Now that we have explored the types of data that AIS common components generate, process, and share with each other, it is time to discuss interoperability types, how interoperability might be supported in AISs, and finally, how interoperability might lead to reuse opportunities across different types of AIS platforms.

2   Examining Interoperability and Reuse in AISs

Santos and Jorge [28] argued that "because of interoperability issues, intelligent tutoring systems [a subset of AISs] are difficult to deploy in current educational platforms without additional work. This limitation is significant because tutoring systems require considerable time and resources for their implementation. In addition, because these tutors have a high educational value, it is desirable that they could be shared, used by many stakeholders, and easily loaded onto different platforms."

At the heart of the problem is a lack of interoperability which is the inability of two or more software-based systems (or components) to cooperate by sharing information in spite of differences in their programming language, interfaces, and functions [5]. Interoperability may be thought of as a scalable form of reuse [5]. Therefore, logic dictates that any enhancement to make systems more interoperable will likely result in higher reuse of components, lower development costs, and increased opportunities for collaboration. For AISs, our goal is to allow plug-and-play applications for the principle components described in the section above: domain, learner/team, instructional, and interface models. If we think of AISs as composable networks, then plug-and-play components are capable of configuring both themselves and other cooperative network components without human intervention. They are in fact intelligent agents. Whether this is possible or practical to support plug-and-play will be discussed in this section. We begin by describing two levels of interoperability.

2.1   Syntactic and Semantic Interoperability

According to Ouksel and Sheth [29] and Euzenat [30], interoperability occurs at two levels:

- Syntactic Interoperability: occurs when two or more systems are able to communicate by exchanging data; syntactic interoperability is a prerequisite for semantic interoperability defined below
- Semantic Interoperability: occurs when the data exchanged between two or more systems is understandable to each system and can be used in each systems processes

In addition to internal component interoperability, we should also consider how AISs and their components might be enabled to be interoperable with non-AIS components? There are in fact many requirements for AIS interoperability:

- Internal component interoperability
- External system and component interoperability
  - Other AISs (e.g., information consumers and producers as part of the Total Learning Architecture [31])
  - Learning Management Systems (LMSs) and other repositories
  - Computer-based Systems (e.g., training simulations)
  - Sensors and other data-generating components

We explore each of these interoperability requirements below in the context of the Generalized Intelligent Framework for Tutoring (GIFT) [32, 33].

2.2   Internal Component Interoperability

In AIS architectures like GIFT, there are standardized messages that facilitate the exchange of information between internal components (e.g., sensor module, learner module, pedagogical module). GIFT's standardized messages provide a form of syntactic and semantic interoperability within GIFT. Sottilare and Brawner [34] identified the following categories of messages (**changes/additions shown in bold**) which might be also be used by other systems to promote internal interoperability and reuse of components from other AISs or AIS architectures:

- Domain Model
  - Inputs
    - Requests for action (from Instructional Model)
      - **Increase/decrease scaffolding (support)**
      - **Increase/decrease frequency or type of feedback**
      - **Increase/decrease the difficulty of future problems or scenarios**
    - Feedback associated with concepts
    - A model of domain tasks, conditions, and standards (measures)
  - Outputs
    - Learner assessments (to Learner Model)
      - **Performance states**
      - **Learning states**
      - **Domain Proficiency states**
      - **Retention models and states**
      - **Transfer of skills states**
      - **Emotional states (moderators of performance and learning states)**
- Learner Model
  - Inputs
    - Learner assessments for each learning objective or concept (from Domain Model)
    - Learner State representation (from Domain Model or derived from data)
    - Sensor data (if applicable)
    - Longer term data (if applicable)

- **Outputs**
  - **Domain Proficiency (to long term learner model at end of course or lesson)**
  - **Learner State representation (to Instructional Model) to support new recommendations or strategies**
- Instructional Model
  - Inputs
    - Learning State representation (from Learner Model)
      - Cognitive state of the learner
      - Performance expectations (above, below, at) for each concept
      - Predicted future performance based on competency model
    - Physiological State representation (from Learner Model)
      - Derived emotional, physical states (e.g., fatigue)
      - Physiological stressors
    - Behavioral State representation (from Learner Model)
      - Derived attitudes or psychomotor performance based on primitive behaviors
    - Longer term learner attributes (from Learner Model or Learner Record Store)
      - Demographics and traits
      - Historical performance (competency **and/or levels of proficiency**)
  - Outputs
    - Request for **changes to** course direction (to Domain Model)
    - Request for feedback (to Domain Model)
    - Request for scenario adaption (to Domain Model)
    - Request for assessment (to Domain Model)

Additional message sets will need to be developed to accommodate the exchange of new information (**in bold**) between AIS components. As suggested by Bell and Sottilare [35] (this volume), intelligent agents can and likely will play a large role in observing the learner and the environment and then triggering interactions (sharing of information) between AIS internal components.

2.3   External System and Component Interoperability

The next target of opportunity for AIS interoperability that we will discuss is AIS compatibility with external systems and components. In GIFT, a standard gateway has been defined to allow external systems and components (e.g., sensors, repositories, and training simulations) to: (1) push data into GIFT to support instructional decisions and assessments and (2) pull data from GIFT to act on either the learner or the environment. This provides a level of syntactic interoperability through the transport of data. Opportunities for semantic interoperability are enabled by defining variables and their data structures so AISs can understand how to use this data within AIS processes. A set of desired instances of interoperable systems/components is discussed in the sections below.

Interoperability with Other AISs

A major standardization goal for AISs is their compatibility with other AISs to the level that major components (e.g., learner models or domain models) from one AIS could be inserted into another AIS and still function appropriately. Within the GIFT architecture, learner models are constructed based upon the variables and measures established by the AIS author. For instance, the AIS author might have a learning objective to master concepts associated with rifle marksmanship. Since this is a psychomotor task, it will be important for the AIS to track learner behaviors associated with steady position, aiming, breathing control, and trigger squeeze. The data would be acquired by sensors and used to assess performance states for steady position, aiming, breathing control, and trigger squeeze in the domain model. The derived performance states would then be transferred to the learner model.

If the task was different (e.g., solving quadratic equations) then the learner model might track whether any of the three steps listed below were completed and in what order:

- Step 1 Divide all terms by a (the coefficient of x2).
- Step 2 Move the number term (c/a) to the right side of the equation.
- Step 3 Complete the square on the left side of the equation and balance this by adding the same value to the right side of the equation.

It might also be desirable to track individual learning during group tasks (e.g., collaborative problem solving) assess contributing factors to successful performance [10]. In this case, we could see the need for a collective model of the group that receives state information from all the supporting AISs relative to their roles, tasks, and progress toward learning objectives for the subject collective task. To automate assessment, the collective model might be tied to an agent-based hierarchical model that decides on a group strategy for task feedback and support.

Interoperability with Learning Management Systems (LMSs)

Another target of opportunity to enhance interoperability in AISs is to enable their functional interaction with learning management systems (LMSs) like edX, Blackboard, Moodle, or Canvas. LMSs are software applications that support the authoring, documentation, tracking, reporting and delivery of instruction for educational courses (e.g., Massive Open Online Courses – MOOCs) and training programs (e.g., Florida Boating Safety) [36]. AISs might be used to stimulate LMSs and provide adaptations (e.g., tailoring of feedback, support, and content). For instance, Aleven and colleagues [37] used GIFT and the Cognitive Tutor to stimulate adaptations in an edX course resulting an adaptive MOOC. The Learning Tools Interoperability (LTI) e-learning standard [38] was used to facilitate the integration of GIFT, the Cognitive Tutor and edX. In this instance, GIFT was an LTI tool provider and edX was an LTI tool consumer as defined in the 2019-1 GIFT documentation:

- LTI Tool Consumer – system that requests an external source for access to their learning tools and then displays it internally
- LTI Tool Provider – system that provides the learning tool to the Consumer

Interoperability with Computer-Based Systems

As noted in the section above about interoperability with LMSs, we also envision the need to be interoperable with other computer-based systems supporting instruction. Training systems that are currently classified as low adaptive systems or systems that only adapt on performance might benefit from additional tailoring based on individual learner or team learning needs, goals, and preferences. For instance, the serious game Virtual BattleSpace (VBS) is a staple for military training in many countries.

VBS has been integrated with GIFT (via the GIFT gateway module) to allow for more comprehensive tailoring. The primary function of the gateway module is to listen for communication outside of GIFT and then convert it into GIFT messages and vice-versa. When a message is received from outside of GIFT (e.g., via the VBS distributed interactive simulation (DIS) connection), the gateway module converts that message into a GIFT message and sends it to the gateway module's topic which is used to send GIFT simulation messages from interop plugins affiliated with systems or available information streams (e.g., DIS, VBS plugin).

A common gateway approach may be a viable method of promoting syntactic interoperability between AISs and other computer-based systems used for training (task learning) and education (concept learning). Associated interop plugins would be required for any system or component that wanted to share information with a gateway-compliant AIS. To achieve semantic interoperability, a mechanism is needed to identify the variables to be shared and the conditions they are intended to assess. In GIFT this is accomplished through a JavaScript condition class. For example, a VBS, a condition class might be used to check whether a specific entity avoided an area in the virtual environment represented in a VBS scenario. This information could then be used to assess the learner's ability to move by terrain association and/or dead reckoning while avoiding certain obstacles, areas or terrain features which is a learning objective for mastering a land navigation task.

Interoperability with Sensors and Other Data-Generating Components

Sensors are devices which detect, identify, acquire, measure, and store a physical property associated with the environment or the behaviors or physiology of the learner (s). Sensors also require gateways to facilitate the movement of data that is acquired by the sensor to GIFT.

3   Recommendations for Interoperability Standards

Our first recommendation is to examine closely what has been done with GIFT that might enhance the internal component interoperability of AISs along with their inter-operability with other systems and components. GIFT was designed to allow instruction in a variety of domains and its architectural principles have enabled interoperability with a variety of systems. The types of interoperability that have been demonstrated by GIFT support effective interaction with systems and components in a way that allows the maintenance of intellectual property by treating the internal processes as black boxes. Interoperability is facilitated at the shared data level.

Our second recommendation is to treat the types of interoperability described in Sect. 2 of this paper as separate requirements. A solution (standard or recommended practice) proposed for one requirement should not affect the type of solutions proposed for other requirements.

Finally, while it is basic to what AISs are, we recommend revisiting the structure that established AISs as four common components. There are still applications of AISs that don't specifically fit this model or processes that are usually in one component that have been shifted to run in other AIS components. Agreeing on a conceptual model at either a component or process level is the precursor to any successful standards or recommended practices that are intended to facilitate the movement of AIS products to the marketplace.

References

1. Sottilare, R., Brawner, K.: Component interaction within the Generalized Intelligent Framework for Tutoring (GIFT) as a model for adaptive instructional system standards. In: The Adaptive Instructional System (AIS) Standards Workshop of the 14th International Conference of the Intelligent Tutoring Systems (ITS) Conference, Montreal, Quebec, Canada, June 2018
2. Anderson, J.R., Franklin Boyle, C., Reiser, B.J.: Intelligent tutoring systems. Science **228** (4698), 456–462 (1985)
3. Psotka, J., Mutter, S.A.: Intelligent Tutoring Systems: Lessons Learned. Lawrence Erlbaum Associates, Hillsdale (1988). ISBN 978-0-8058-0192-7
4. Baylor, A.: Beyond butlers: intelligent agents as mentors. J. Educ. Comput. Res. **22**(4), 373–382 (2000)
5. Wegner, P.: Interoperability. ACM Comput. Surv. (CSUR) **28**(1), 285–287 (1996)
6. Sottilare, R.: Considerations in the development of an ontology for a Generalized Intelligent Framework for Tutoring. In: International Defense & Homeland Security Simulation Workshop in Proceedings of the I3M Conference, Vienna, Austria, September 2012
7. Fletcher, J.D. Sottilare, R.: Shared mental models of cognition for intelligent tutoring of teams. In: Sottilare, R., Graesser, A., Hu, X., Holden, H. (eds.) Design Recommendations for Intelligent Tutoring Systems. Learner Modeling, vol. 1. Army Research Laboratory, Orlando (2013). ISBN 978-0-9893923-0-3
8. Sottilare, R.: Adaptive Intelligent Tutoring System (ITS) Research in Support of the Army Learning Model - Research Outline. Army Research Laboratory (ARL-SR-0284), December 2013
9. Sottilare R., Ragusa C., Hoffman, M., Goldberg, B.: Characterizing an adaptive tutoring learning effect chain for individual and team tutoring. In: Proceedings of the Interservice/Industry Training Simulation and Education Conference, Orlando, FL, December 2013
10. Sottilare, R.A., Burke, C.S., Salas, E., Sinatra, A.M., Johnston, J.H., Gilbert, S.B.: Designing adaptive instruction for teams: a meta-analysis. Int. J. Artif. Intell. Educ. (2017). https://doi.org/10.1007/s40593-017-0146-z
11. Sottilare, R., Graesser, A., Hu, X., Sinatra, A.M.: Introduction to team tutoring and GIFT. In: Design Recommendations for Intelligent Tutoring Systems. Team Tutoring, vol. 6. U.S. Army Research Laboratory, Orlando (2018). ISBN 978-0-9977257-4-2
12. Mitrovic, A., Martin, B., Suraweera, P.: Intelligent tutors for all: the constraint-based approach. IEEE Intell. Syst. **4**, 38–45 (2007)

13. Anderson, L.W., et al.: A Taxonomy for Learning, Teaching, and Assessing: A revision of Bloom's Taxonomy of Educational Objectives. Pearson, Allyn & Bacon, New York (2001)
14. Sottilare, R., Holden, H., Brawner, K., Goldberg, B.: Challenges and emerging concepts in the development of adaptive, computer-based tutoring systems for team training. In: Proceedings of the Interservice/Industry Training Simulation & Education Conference, Orlando, Florida, December 2011
15. Sottilare, R.A., Burke, C.S., Salas, E., Sinatra, A.M., Johnston, J.H., Gilbert, S.B.: Designing adaptive instruction for teams: a meta-analysis. Int. J. Artif. Intell. Educ. **28**(2), 225–264 (2018)
16. Johnston, J., Sottilare, R., Sinatra, A.M., Burke, C.S. (eds.): Building Intelligent Tutoring Systems For Teams: What Matters. Emerald Group Publishing, Bingley (2018)
17. Salas, E.: Team Training Essentials: A Research-Based Guide. Routledge, London (2015)
18. Aleven, V., Mclaren, B., Roll, I., Koedinger, K.: Toward meta-cognitive tutoring: a model of help seeking with a Cognitive Tutor. Int. J. Artif. Intell. Educ. **16**(2), 101–128 (2006)
19. Rus, V., Arthur, C.G.: The question generation shared task and evaluation challenge. The University of Memphis, National Science Foundation (2009)
20. Graesser, A.C.: Conversations with AutoTutor help students learn. Int. J. Artif. Intell. Educ. **26**(1), 124–132 (2016)
21. Johnson, W.L., Lester, J.C.: Face-to-face interaction with pedagogical agents, twenty years later. Int. J. Artif. Intell. Educ. **26**(1), 25–36 (2016)
22. Nye, B.D., Graesser, A.C., Hu, X.: AutoTutor and family: a review of 17 years of natural language tutoring. Int. J. Artif. Intell. Educ. **24**(4), 427–469 (2014)
23. D'Mello, S.K., Graesser, A., King, B.: Toward spoken human–computer tutorial dialogues. Hum. Comput. Interact. **25**(4), 289–323 (2010)
24. Litman, D.: Natural language processing for enhancing teaching and learning. In: Thirtieth AAAI Conference on Artificial Intelligence, 5 March 2016
25. Baker, R., D'Mello, S., Rodrigo, M., Graesser, A.: Better to be frustrated than bored: the incidence and persistence of affect during interactions with three different computer-based learning environments. Int. J. Hum. Comput. Stud. **68**(4), 223–241 (2010)
26. D'Mello, S., Graesser, A.: Dynamics of affective states during complex learning. Learn. Instr. **22**(2), 145–157 (2012)
27. Goldberg, B.S., Sottilare, R.A., Brawner, K.W., Holden, H.K.: Predicting learner engagement during well-defined and ill-defined computer-based intercultural interactions. In: D'Mello, S., Graesser, A., Schuller, B., Martin, J.-C. (eds.) ACII 2011, Part I. LNCS, vol. 6974, pp. 538–547. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-24600-5_57
28. Santos, G.S., Jorge, J.: Interoperable intelligent tutoring systems as open educational resources. IEEE Trans. Learn. Technol. **6**(3), 271–282 (2013)
29. Ouksel, A.M., Sheth, A.: Semantic interoperability in global information systems. ACM SIGMOD Rec. **28**(1), 5–12 (1999)
30. Euzenat, J.: Towards a principled approach to semantic interoperability. In: Proceedings of IJCAI 2001 Workshop on Ontology and Information Sharing, pp. 19–25, 4 August 2001. No commercial editor
31. Folsom-Kovarik, J.T., Raybourn, E.M.: Total Learning Architecture (TLA) enables next-generation learning via meta-adaptation. In: Interservice/Industry Training, Simulation, and Education Conference Proceedings, November 2016
32. Sottilare, R.A., Brawner, K.W., Goldberg, B.S., Holden, H.K.: The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. U.S. Army Research Laboratory—Human Research & Engineering Directorate (ARL-HRED), Orlando, FL, USA (2012)

33. Sottilare, R., Brawner, K., Sinatra, A., Johnston, J.: An updated concept for a Generalized Intelligent Framework for Tutoring (GIFT). US Army Research Laboratory, Orlando, FL, USA (2017)
34. Sottilare, R., Brawner, K.: Exploring standardization opportunities by examining interaction between common adaptive instructional system components. In: Proceedings of the First Adaptive Instructional Systems (AIS) Standards Workshop, Orlando, Florida, March 2018
35. Bell, B, Sottilare, R.: Adaptation vectors for instructional agents. In: Proceedings of the First HCII Adaptive Instructional Systems Conference, Orlando, Florida, July 2019
36. Ellis, R.K.: Field guide to learning management systems. ASTD learning circuits, 1–8 August 2009
37. Aleven, V., Sewall, J., Andres, J.M., Sottilare, R., Long, R., Baker, R.: Towards adapting to learners at scale: integrating MOOC and intelligent tutoring frameworks. In: Proceedings of the Fifth Annual ACM Conference on Learning at Scale, p. 14. ACM, 26 June 2018
38. IMS Global. Learning Tools Interoperability (LTI) Version 1.2 (2019). https://www.imsglobal.org/specs/ltiv1p2