Practical Requirements for ITS Authoring Tools from a User Experience Perspective

Scott Ososky(✉)

U.S. Army Research Laboratory (ARL), Orlando, FL, USA
Scott.J.Ososky.ctr@mail.mil

Abstract. Intelligent Tutoring Systems (ITS) are not yet widely implemented in learning, despite the general prevalence of digital resources in educational and training environments. ITS have been demonstrated to be effective for learners, but ITS development is not yet efficient for authors. Creating an ITS requires time, resources, and multidisciplinary skills. *Authoring tools* are intended to reduce the time and skill required to create an ITS, but the current state of those tools is categorized as a series of design tradeoffs between functionality, generalizability, and usability. In practice, the former two factors matter little if potential authors disregard the ITS in favor of other solutions. In this sense, *authors*, not learners, are the primary users of an ITS; the user experience of authors is critical to greater ITS adoption at an organizational level. With those challenges in mind, ongoing work and lessons learned on the design of authoring tools are described for a specific ITS platform, the Generalized Framework for Intelligent Tutoring (GIFT). User-centered design considerations are examined through the lens of authors' goals, mental models for authoring, and the definition of authoring sub-roles. Recommendations for authoring tool design and future research directions for design research in authoring tools are discussed.

Keywords: Intelligent tutoring systems · Adaptive tutoring · Authoring tools · User-centered design · User experience · Mental models · Design research

1 Introduction

Intelligent Tutoring Systems (ITS), or adaptive tutors, are learning systems that have the ability to collect data about a learner, including assessments, attributes, and states, in order to tailor content to that learner's needs. These systems have been demonstrated to be more effective than many other types of instruction, approaching the effectiveness of one-to-one human tutoring [1]. However, ITS are not yet widely implemented in educational [2, 3] or military [4] environments, despite the prevalence of general digital resources in learning environments. One contributing factor to the slow adoption of ITS is the lack of accessible authoring tools that can meet the needs of modern training and educational institutions. There has been an extensive body of research on the engineering and development of authoring tools, but there is a considerable lack of literature related to user experiences and user goals associated with authoring tools. Thus, the current

© Springer International Publishing Switzerland 2016
D.D. Schmorrow and C.M. Fidopiastis (Eds.): AC 2016, Part II, LNAI 9744, pp. 55–66, 2016.
DOI: 10.1007/978-3-319-39952-2_6

work takes a user-centered approach to authoring tool design and user acceptance. End-user needs are also discussed in the context of identifying requirements for and focusing engineering efforts in authoring tools. First, an overview of ITS as well as learning and training environments is provided.

1.1 Intelligent Tutoring Systems

ITS, which are also referred to as adaptive tutors, have been described as "computer-based learning systems which attempt to adapt to the needs of learners" (p. 350) [5], or a computer system that customizes instruction and/or feedback to learners [6]. Intelligent tutoring systems are modular with four primary components: a learner/student model, a pedagogical/instructional model, a domain knowledge model, and a user interface/communications model. Each of those components exchanges data with one another, either directly or indirectly. Unlike traditional computer based training, ITS capture and process data about the learner, in order to tailor instruction toward individual learners' needs. In addition to performance and behavioral learner modeling, ITS can support the affective and motivational needs of the learner as well [4, 7].

The educational and training potential within ITS is underscored by one inherent problem, succinctly described by Murray, "Intelligent Tutoring Systems are highly complex educational software applications used to produce highly complex software applications" [8]. Despite decades of research and development, barriers to greater ITS adoption from an authoring perspective include high development costs, limited reuse of tutors, and the multidisciplinary skills required to build a robust tutor [9, 10]. Thus, by focusing on the needs of those that develop and manage adaptive training content, the ITS community can begin to address some of the barriers to greater adoption, both within the classroom and beyond.

1.2 Technology in the Classroom

A recent report from the Bill and Melinda Gates foundation describes the current state of educational technology in the classroom: Although 93 % of participating teachers use some form of digital resources in their instructional plans, more time is spent without digital tools than with them for independent practice, for assessment, and for individual tutoring [2]. Further, 58 % of teachers found digital tools to be effective, but gaps in these tools were reported to exist in all subject areas and grades. For instance, only 33 % of teachers believed that digital tools could be useful in remediation. ITS have the inherent capabilities to support all of those activities.

So why would a teacher select some other instructional method over using an ITS? The report provided some additional clues: When selecting instructional resources, teachers place the most emphasis on cost-effectiveness, ease of integration with their current methods and materials, and the ability of the tool to help teachers tailor instruction toward student needs. Additionally, organizational commitment to technology usage varies wildly, leaving many teachers (31 %) to acquire digital tools and manage the data produced by digital tools on their own [2]. Ultimately, most classroom time is still spent on whole-classroom instruction, despite the availability of digital tools.

Barriers to incorporating ITS (and digital tools, in general) are further compounded for classrooms in developing countries with respect to availability of hardware, infrastructure, and appropriate instructional content [11].

1.3 Tutoring Beyond the Classroom and in Military Domains

Considerations for ITS in developing countries may seem ancillary to tutor authoring concerns, given general availability of internet, electricity, and data connections for the ubiquitous devices in developed nations. However, those barriers which inhibit the use of digital tools in developing world classrooms are not unlike the challenges associated with learning in the field, particularly military training for and in operational environments (sometimes referred to as tutoring in-the-wild).

The Army, for instance, requires training solutions that are cost-effective, readily available and easily accessible. Current Army training needs include leadership, long-term learning, and operational adaptability under conditions of uncertainty and complexity [12]. To that end, there are cases in which training at a desktop computer is not practical, such as training for psychomotor tasks and field work. Further, Soldiers need to be able to train anywhere, at any time, and for a variety of operational constraints. Those constraints (whether simulated for training purposes or actual constrains imposed by the setting) include environments that are noisy, lack a data connection, and may be GPS-denied; a land navigation is one such task in which field training with an ITS is challenging [13]. Soldiers sensory attention and gear configuration in a hemorrhage control training task, for example, may prohibit tactile interaction with a digital tutor [14]. Similar lists of constraints may be identified for other physical and field-based tasks in non-military domains as well.

Finally, ITS (or adaptive tutoring) in military domains is also limited due to the complexity and ill-defined nature of military tasks. For instance, there are many aspects to peacekeeping operations in foreign environments, where the structure of knowledge is not declarative, there are no clear right or wrong solutions, and the quality of those solutions are likely to change based on specific situational factors [4]. Task complexity and ill-defined tasks are challenges to ITS adoption in the classroom and the field, in military applications and elsewhere (e.g., a tutor for jazz piano, or theatrical improvisation).

2 Authoring Tools: From Usability to User Experiences

The previous section paints a broad picture of the potential for ITS in educational and training settings along with real-world factors that inhibit increased ITS use. Greater use of adaptive tutoring in training and education ultimately requires the trust and confidence of training and educational stakeholders. Students and learners might be considered stakeholders, but they typically consume the courses developed within an ITS. Thus, the stakeholders of current interest are course creators, data managers, and facilitators. Teachers, instructional designers, and subject matter experts may fill one or more of these roles; therefore the needs of these users are paramount in designing sub-systems

and tools that foster trust and confidence in ITS. Authoring tools for ITS do not directly address all of the needs of those groups, but authoring tools influence and are influenced by every other aspect of the ITS, including the tutor-user interface, pedagogical engines, domain models, and data management dashboards.

Much has been written on the topic of theory- and engineering-based efforts to develop authoring tools that provide to potential authors the functions necessary to create tutors without computer science or instructional design knowledge [15–18]. Murray, specifically, has published numerous works on authoring tools over the past two decades. His work includes analysis of the problem (or opportunity) in which design tradeoffs are made in authoring tools between usability, depth, and flexibility [19, 20]. In summary, increasing the power of the authoring tools (i.e., depth), the applicability of the tools to different domains and problem spaces (i.e., flexibility), or the usability of the tools themselves (i.e., learnability, productivity), comes at a cost to one or both of the other two [3].

The current discussion does not endeavor to duplicate the effort of those prior works, rather to extend and build upon the dialogue on tradeoffs in two very important ways from the perspective of user experience (UX) research and practice.

2.1 Authoring Tools and the Reality of Alternatives

First, it is imperative to start thinking about ITS and Authoring Tools from a product perspective if the ITS community truly wants to have a user base that extends far beyond its respective ITS research communities. Even if monetization is not part of an ITS project plan, the non-learner users of an ITS must be thought of and treated like customers. For instance, any time teachers, subject matter experts, or instructional designers interact with a channel associated with an ITS product (e.g., software, website, documentation), those users' experiences should be ones in which they feel confident and positive about using the associated authoring tools. Further, regardless of users' skill levels, developers should aspire to create experiences that make users feel smart when using their authoring tools. Consider, for example, how patient and tolerant a potential user might be with a set of authoring tools, before giving up completely. As the idiom states, you only have one chance to make a first impression.

Taking a product perspective allows ITS development teams to consider the reality of alternatives. There are currently a handful of ITS systems in the public domain, which are a subset of available digital tools in general, in addition to tools already available in the analog space. Simply put, a potential user of an ITS can consider many alternatives whose functionality and learning-benefit may vary wildly in comparison to a specific ITS. However, if a textbook, digital tool, other ITS, field manual, job-aid, etc. is perceived by a potential author to be *good enough*, the usability-depth-flexibility challenge becomes something of a moot issue. In some cases, a decision to not use an ITS may result in a potential author choosing to do *nothing at all,* focusing their time elsewhere. Recall, the reasons why ITS have not been more widely adopted in the first sections of this paper. Marketing might contribute to the problem of ITS adoption, but marketing is outside the scope of this discussion. While superficially similar, marketing is *selling*, user experience is *serving* [21].

This is not to say that the depth and flexibility of the product are not important; of course they are. Great experiences can quickly sour if there is no substance to the product. Rather, the emphasis on UX complements Murray's notion that subject matter experts (i.e., potential authors) should be involved in the ongoing development of authoring tools [3, 10]. Identifying user goals, user expectations, and the issues that users encounter provides a path to creating *authoring tools right*, in addition to the *right authoring tools*. Furthermore, the overall user experience associated with ITS and its authoring tools need to not just be *as good* as whatever a potential ITS user is doing now, but it has to be *better* in order to justify switching to a new method of training or educating. This applies to the actual author as well as their associated organizational stakeholders. In that sense, the concept of the positive user experience covers all types of authoring systems along the usability-depth-flexibility continuum.

2.2 Complex Authoring Tools Can Be Usable

Second, UX design can help the ITS community to rethink the problem of trading authoring tool depth or flexibility for usability. Murray operationalized usability for ITS Authoring Tools as *learnability* (how easy is it to learn a system) and *productivity* (how efficiently a tutor can be authored) [3]. In order to rethink the usability tradeoff, inspiration can be found within the similarly complex domain of game design and their associated game development tools.

Like intelligent tutors, games come in many shapes and sizes, with various levels of depth and flexibility. With respect to learnability, the barrier to game design previously limited the user-base to those with computer programming experience; today, freely available development tools and game engines provide the foundation for designers of all skill levels to start learning to create their own games (again, of varying scope and complexity). The powerful, flexible Unity and Unreal game development tools, respectively, are available at no cost, whose learnability is broadly supported by user-driven communities. The popular Minecraft game is an example of using a simple tool to teach students about a variety of subjects including programming fundamentals [22]. There are also genre-specific game development tools including Super Mario Maker and RPG Maker, for 2D-platform and role-playing games, respectively.

Similarly, the potential for what an ITS could be is still evolving, which creates a moving target for authoring tool design [19]. There will continue to be a variety of ITS platforms with authoring tools of varying levels of depth and flexibility corresponding to the needs of various learning environments. Along the depth/flexibility continuum, the key is providing authoring tools with a comprehensive approach to learnability including forums, examples, tutorial videos, and web-documentation, in addition to usable tools. In this way, a novice author that uses simple authoring tools to create a simple tutor can gain the knowledge and confidence to use complex authoring tools to create a complex tutor in the future, apply skills to a different authoring platform, or perhaps develop an entirely new type of tutor, not yet imagined.

With respect to productivity, current user experience research in game development tools seeks to increase the efficiency of even the most complex development tools, for example, by reducing the time to complete specific tasks, helping the user to organize

information, and reducing the potential for user error [23]. These are usability improvements that do not come at the cost of depth and flexibility; rather research effort improves development tools by identifying user goals, understanding human mental models of specific tasks, and adjusting interface elements in order to help developers to accomplish their tasks in a more efficient way. Similar efforts are needed to increase the efficiency of ITS authoring tools, without changing the power of the existing system. The power of specific authoring tools can also scale with automation and templates. In this way, simple and complex authoring tools become part of the same system, giving the author the flexibility to discover and explore advanced authoring functions, when desired (see [10] for an authoring tool design case study).

In thinking about the design of authoring tools, the ITS community should endeavor to separate usability from the depth and flexibility tradeoff. Next, the notion of usability should be expanded to a comprehensive user experience for authoring tools, which considers every point of interaction between the system and the potential user. This approach applies to authoring tools that have both limited and wide application, as well as tools that exist today and ones not yet conceived. Because user trust is slowly gained, and easily lost, the needs of potential authors and organizational stakeholders should be at the center of any authoring tool design plan. If it seems difficult to justify allocating resources to the UX of authoring tools, consider the cost of *not* doing this work with respect to the user base and overall success of the ITS platform.

3 Practical UX Requirements for Authoring Tools

The current work concludes with some practical examples of in-progress user experience efforts for a specific ITS platform, the Generalized Intelligent Framework for Tutoring (GIFT). GIFT is described as "an empirically-based, service-oriented framework of tools, methods, and standards to make it easier to author computer-based tutoring systems (CBTS), manage instruction and assess the effect of CBTS, components and methodologies" [4]. Simultaneously, GIFT is an open-source research project and public-facing application. GIFT is currently under development and includes a number of technologies, features, tools, and methods intended to support a variety of users including instructional designers, authors, instructors, researchers, and learners.

While the UX requirements described below are framed in the context of GIFT-based research, the requirements can be adapted to other ITS authoring tools. Finally, this list is not intended to be comprehensive. Instead, the examples in the following sections serve to illuminate UX concerns and generate dialog for improving authoring tools to the benefit of current and future adaptive tutor authors and associated organizations.

3.1 Comprehensive Help and Documentation

GIFT has seen numerous improvements to its authoring tools and overall authoring workflow. In the absence of formal tools, GIFT courses were created by editing eXtensible markup language (XML) files. The first formal authoring tools allowed for indirect manipulation of the XML files through desktop software. Currently, web-based

applications are being developed that provide a more menu-based approach to creating courses. While the underlying structure of the GIFT architecture has not changed dramatically, the end-user authoring experience has seen significant change. In order to provide a consistent experience with the web-application, the GIFT team is working to transition the support documentation to an online, web-based format, which can be rapidly updated along with the application.

ITS Documentation and help files must (first exist, and then) keep pace with authoring tool development. Creating this material may be one of the last tasks in a development effort, but documentation will be one of the first elements that an ITS author will seek when encountering an issue. Therefore documentation warrants considerable attention. Documentation should be up to date, searchable, and internally cross-referenced. Help (in the form of descriptions, hints, and tips) should be easily accessible within the interface at the point-of-need. Help and documentation should serve to support authors' knowledge and confidence, as well as prevent them from getting *lost* in authoring tool interfaces.

3.2 Beyond Documentation: Demonstrations and Social Channels

ITS authoring tools should contain example courses to inspect and modify. These examples should be highly polished (both visually and functionally) in order to make a positive first impression on users. To the extent that is practical, supplementary documentation should reference these examples in order to ground the authors' knowledge in a tangible work. Complementary to that, ITS support should extend beyond software tools into social media, which might include tutorial videos, conversations in forums, and interactions with users via social channels. Some of the topics to discuss in these channels are obvious: What is [product name]? What do I do now/first? Or, what is a tutor and how is one created? Topics for other tutorials and discussions can be generated from user research, as well as through internal discussion within the interdisciplinary development team. Finally, opening channels for conversation will help authors to connect one another, and hopefully form a community around which a user base can grow and improve together.

GIFT, for example, has forums in which users can quickly connect with key members of the development team. Additionally, the GIFT authoring tools contain a number of courses that demonstrate technical functionality. These courses can also be viewed within the authoring tools to get a sense of how those courses were created. However, much of the burden is on the author to discovering how all of the systems work together, and it is not immediately clear how these examples would apply to authors' original course creations. The GIFT team is currently working to develop additional example courses, as well as provide additional support through interactions in forums and via tutorial videos in order to support knowledge development.

3.3 Authoring for Non-traditional Learning Environments

When designing a tutor, authors must consider their learners and the environments in which they will interact with the tutor. Thus, authoring system design must also consider

the learning contexts for which authors will need appropriate tools in order to properly configure a *tutor-user interface*. For example, authoring systems may need visual styling options in order to simultaneously support PC, tablet, and mobile displays, respectively. Where text input may be appropriate for PC learning, voice input may be required for tutoring on a mobile device in a field environment. Further outside-the-box thinking might consider tutoring with wearable displays such as smart watches, smart glasses, or augmented reality headsets in which novel methods are needed to communicate across the tutor-user interface. Other considerations include designing courses for limited power (processing or battery, respectively) capacity as well as supporting downloadable courses to use in a temporarily offline state.

Multiple lines of GIFT research and development are addressing mobile, offline, and field tutoring, respectively. The GIFT Cloud web-application provides the first steps toward browser-based compatibility with mobile and tablet systems. The PC-based GIFT Local supports offline use, with a server-based implementation in planning for secure network environments. Finally, distinct research efforts are examining non-traditional displays, such as sand-tables [24] and smart-glasses [13, 14], in order to define software and hardware requirements supporting tutoring *in-the-wild* for military tasks in operational settings. These efforts, in turn, continue to define requirements for new authoring tools including, for example, situated GIFT authoring within an external software application.

3.4 Collaboration, Sharing, and Authoring Roles

Part of a comprehensive user experience is understanding how actual users might use an ITS in a real-world environment. Given the skills involved in creating a tutor (e.g., computer science, instructional design, and subject matter expertise) it is reasonable to anticipate that *teams of authors*, instead of individual authors, will be creating adaptive tutors. These teams might be centrally located or geographically distributed. They may be working on the same course at the same time or multiple courses simultaneously. As such, ITS authoring tools will need to have features supporting, for instance, collaborative editing, change tracking, approval, and version control. Additionally, user roles will be necessary to support project management structures within the organization using the tools. Authoring roles can benefit specialized authors as well; for example, the tools and default view available to a subject matter expert should be different than those of an instructional designer.

GIFT, specifically, is research-driven. Thus, GIFT has a particular need to support experimental research, in addition to collaborative authoring. To that end, additional tools have been created in order to facilitate experimentation in topics related to adaptive tutoring. While those tools have no direct impact on the usability, depth, and flexibility of the authoring tools, the research tools complement the authoring tools in a way that provides functionality that meets needs of a subset of end-users. Specialized interfaces allow researchers to create research studies from existing courses, with specialized access links, and participant data reporting interfaces. The GIFT team continues to gather information about how authors might want to utilize GIFT, and consider this input for UX and feature improvements in subsequent platform updates.

3.5 Mental Models for ITS and Authoring Tools: The Elephant in the Room

Finally, Murray [19] explained that authoring tools should help users build accurate mental models of the ITS building blocks, configurations, and workflow afforded by the authoring tool. This is inherently difficult, because ITS are evolving, and each ITS will differ in some ways from others. However, mental model theory can provide some guidance with respect to approaching this interaction problem en route to an *accurate* mental model. Rouse and Morris [25] described mental models as "mechanisms whereby humans are able to generate descriptions of purpose and form, explanations of system functioning and observed system states, and predictions of future states" (p. 7). Mental models influence users' expectations regarding a system's functionality and guide user interaction behavior [26]. An individual's mental model regarding a particular system is influenced by past experiences and perceived similarity of other systems to the target system. Further, human mental models do not need to be complete or even accurate to be applied to a specific system interaction [27].

To that end, there are a number of approaches for designing authoring tools with human mental models in mind. For instance, in the absence of a mental model for ITS authoring, users will attempt to leverage a known model of another system, and test assumptions about the authoring tools, based on that model. The mental model that a user selects can be influenced by the look and feel of the authoring tool. For instance, a mental model of PowerPoint may be suggested by stacking course elements along the left side of the UI, with the design space occupying a larger right-side area. The use of metaphors and existing mental models can be useful in acclimating new users to a system. However, the usefulness of that approach has limits; and in order to leverage a metaphor it is generally necessary to also understand the user's mental model of the metaphor system (e.g., what they know about PowerPoint). The goal of this approach is that, over time, users will exhaust the metaphor model and develop a distinct mental model of the target authoring system.

An alternative approach leverages general and specific mental models in helping new users to understand ITS authoring tools. Generally, there are some basic concepts and components that are common to ITS (e.g., learner model, pedagogical model, domain knowledge, tutor-user interface). However, the specific manner in which these components are operationalized will differ with each ITS, and therefore within each authoring system. In order to address the learnability of a specific authoring system, it may beneficial to first establish the general mental model of an ITS with potential authors. Then, potential authors can use this generalized mental model to develop a specific model of a particular ITS and associated authoring tools. In this way, authors would be better prepared to move between novice and expert tools within the same authoring system, or apply their knowledge to other authoring systems, with a stronger baseline mental model for adapting to those new experiences.

GIFT, specifically, has a variety of general (theory-based) and specific (applied) scholarly publications located on the web-based portal located at www.gifttutoring.org in order to support current and potential users in their development of system mental models. Additionally, GIFT has seen significant improvements to the authoring interface with the alpha release of the web-based GIFT Cloud application. Current efforts are

examining the use of familiar interface elements to potentially evoke system metaphors and increase the learnability of GIFT. Potential authoring design interfaces being considered include lists and hierarchical structures, as well as object-oriented structures such as flow charts and discrete-event models.

4 Discussion

Intelligent tutoring systems have the potential to revolutionize how, when, and where learners can interact with instructional content. Though, despite decades of research and development in ITS, they are not yet widely used outside of research and development settings. Significant barriers to the greater adoption of ITS in educational and training contexts include the steep learning curve and high resources required to create adaptive tutors. Authoring tools can reduce these barriers and make ITS more accessible to both novice and expert adaptive tutor authors. While authoring tools continue to evolve in depth and flexibility, it is necessary to expand the concept of tool usability toward a comprehensive end-to-end user experience with respect to actual users interacting with ITS authoring products in real-world contexts.

Interactions with ITS authoring tools should build trust and confidence in their users, and authors should feel smart when interacting with authoring tools. Authors should be able to view high-quality example courses showcasing the power and/or flexibility of an ITS, and then use authoring tools to examine and deconstruct those examples. Authoring tools should provide capabilities that allow novice users to create tutors, while providing discoverable advanced functions that help novice users become experts. Further, simple authoring tools for simple tutors create opportunities for potential users to get interested in tutor authoring. Forums, social media, and video tutorials are recommended to grow community of practice around the authoring tools, and encourage interaction among members and ITS developers, respectively.

Designing the user experience of authoring tools is not an effort that should come at the cost of theory and platform development; it should be an integral part of the development plan from the outset of an ITS project. By considering the needs, goals, and mental models of potential end-users, system developers create a path to building the right authoring tools, while building authoring tools right. In order to gain greater adoption in educational and training communities, ITS and authoring tool end-user experiences must not just be as good as digital and analog tool alternatives, they must inspire trust and confidence in order to switch from alternatives, they must be better.

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

References

1. VanLehn, K.: The relative effectiveness of human tutoring, intelligent tutoring systems and other tutoring systems. Educ. Psychol. **46**, 197–221 (2011)
2. Bill & Melinda Gates Foundation: Teachers know best: What educators want from digital instructional tools 2.0 (2015). http://collegeready.gatesfoundation.org/2015/12/what-educators-want-from-digital-tools-2-0/
3. Murray, T.: Design tradeoffs in usability and power for advanced educational software authoring tools. Educ. Technol. **44**, 10–16 (2004)
4. Sottilare, R.A., Graesser, A.C., Hu, X., Holden, H.: Design Recommendations for Intelligent Tutoring Systems. Learner Modeling, vol. 1. U.S. Army Research Laboratory, Orlando (2013)
5. Self, J.: The defining characteristics of intelligent tutoring systems research. Int. J. Artif. Intell. Educ. (IJAIED) **10**, 350–364 (1998)
6. Psotka, J., Massey, L.D., Mutter, S.A.: Intelligent Tutoring Systems: Lessons Learned. Psychology Press, Hillsdale (1988)
7. Beck, J., Stern, M., Haugsjaa, E.: Applications of AI in education. Crossroads **3**, 11–15 (1996)
8. Murray, T.: Coordinating the complexity of tools, tasks, and users: on theory-based approaches to authoring tool usability. Int. J. Artif. Intell. Educ. **26**(1), 37–71 (2016)
9. Picard, R.: Building an affective learning companion. Keynote Address at the 8th International Conference on Intelligent Tutoring Systems (2006). http://www.its2006.org/ITS_keynote/ITS2006_01.pdf
10. Murray, T., Woolf, B., Marshall, D.: Lessons learned from authoring for inquiry learning: a tale of authoring tool evolution. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 197–206. Springer, Heidelberg (2004)
11. Nye, B.D.: Intelligent tutoring systems by and for the developing world: a review of trends and approaches for educational technology in a global context. Int. J. Artif. Intell. Educ. **25**, 177–203 (2014)
12. U.S. Army Training and Doctrine Command: The United States Army Learning Concept for 2015, Fort Monroe, VA (2011)
13. Sottilare, R.A., LaViola, J.: Extending intelligent tutoring beyond the desktop to the psychomotor domain. In: Interservice/Industry Training Simulation & Education Conference, Orlando, FL, December 2015
14. Sottilare, R., Hackett, M., Pike, W., LaViola, J.: Adaptive instruction for medical training in the psychomotor domain. In: Cohn, J., Fitzhugh, D., Freeman, H. (eds.) Special Issue: Modeling and Simulation Technologies to Enhance and Optimize the DoD's Medical Readiness and Response Capabilities of the Journal for Defense Modeling & Simulation (JDMS) (2016, in review)
15. Olsen, J.K., Belenky, D.M., Aleven, V., Rummel, N.: Intelligent tutoring systems for collaborative learning: enhancements to authoring tools. In: Lane, H., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS, vol. 7926, pp. 900–903. Springer, Heidelberg (2013)
16. Aleven, V., Sewall, J.: Hands-on introduction to creating intelligent tutoring systems without programming using the cognitive tutor authoring tools (CTAT). In: Proceedings of the 9th International Conference of the Learning Sciences, vol. 2, pp. 511–512. International Society of the Learning Sciences (2010)
17. Suraweera, P., Mitrovic, A., Martin, B.: Widening the knowledge acquisition bottleneck for constraint-based tutors. Int. J. Artif. Intell. Educ. (IJAIED) **20**, 137–173 (2010)
18. Mitrovic, A., Martin, B., Suraweera, P., Zakharov, K., Milik, N., Holland, J., McGuigan, N.: ASPIRE: an authoring system and deployment environment for constraint-based tutors. Int. J. Artif. Intell. Educ. **19**, 155–188 (2009)

19. Murray, T.: Theory-based authoring tool design: considering the complexity of tasks and mental models. In: Sottilare, R.A., Graesser, A.C., Hu, X., Brawner, K. (eds.) Design Recommendations for Intelligent Tutoring Systems. Authoring Tools and Expert Modeling Techniques, vol. 3, pp. 9–29. U.S. Army Research Laboratory, Orlando (2014)
20. Murray, T.: Having it all, maybe: design tradeoffs in ITS authoring tools. In: Frasson, C., Gauthier, G., Lesgold, A. (eds.) ITS 1996. LNCS, vol. 1086, pp. 93–101. Springer, Heidelberg (1996)
21. Fraco, A.: UX vs. Marketing: Can these opposites attract? Agency Post. HubSpot (2014). http://blog.hubspot.com/agency/ux-vs-marketing-opposites-attract
22. Kastrenakes, J.: Microsoft announces Minecraft: Education Edition for schools. The Verge (2016). http://www.theverge.com/2016/1/19/10788994/minecraft-education-edition-announced-microsoft
23. Lightbown, D.: Designing the User Experience of Game Development Tools. CRC Press, Boca Raton (2015)
24. Boyce, M.W., Amburn, C.R., Sottilare, R.A., Goldberg, B.S., Moss, J.D.: The Effect of Topography on Learning Military Tactics. Department of Defense Human Factors and Engineering Technical Advisory Group, Orlando (2015)
25. Rouse, W.B., Morris, N.M.: On looking into the black box: prospects and limits in the search for mental models. Psychol. Bull. **100**, 349 (1986)
26. Ososky, S.: Influence of Task-Role Mental Models on Human Interpretation of Robot Motion Behavior. University of Central Florida, Orlando (2013)
27. Norman, D.A.: Cognitive engineering. In: Norman, D.A., Draper, S.W. (eds.) User Centered System Design: New Perspectives on Human-Computer Interaction, pp. 31–61. Lawrence Erlbaum Associates, Hillsdale (1986)