Could the Technology for Adaptive Learning Systems Come out of GBL?

David Moffat[1], Olga Shabalina[2] and Aleksander Khairov[2]

[1]Glasgow Caledonian University, UK
[2]Independent Researcher

d.c.moffat@gcu.ac.uk
o.a.shabalina@gmail.com
sasha-hairov@mail.ru

Abstract: Games based learning (GBL) has a related field that it can draw upon: intelligent tutoring systems (ITS). A common concern in both fields is *adaptivity*, whereby the system can automatically adapt to the user. In order to support this adaptation, an ITS will generally include a user model and may also have a formal domain model. Work in this area started optimistically some years ago, but seems to have either lost some of that initial enthusiasm or been diverted into other directions. We scan recent ITS literature to help consider why this might be, and suggest how GBL may be the field best placed to take the work forward again. Learner models have long been seen as useful for adaptive learning systems. They include information about the learner which allows the system to adapt the course of learning materials and exercises to the learner's particular characteristics. In order to achieve a good quality of adaptation to the user, a detailed model of the required domain knowledge is typically added. The user and domain models then have to be brought together to lay out a course of exercises for the learner to do, and to track progress as the knowledge is learned. It's an attractive research programme, but recent work has moved to new issues, such as MOOCs. The reasons for that are partly opportunistic and economical, but also suggest a deeper problem with the research programme. It is a costly task to develop a domain model, and a suitable learner model that can take advantage of it. We suggest that GBL is in a good position to push through this cost barrier, because much of the effort is already implicitly involved in the game design process, which typically has to be more rigorously planned out in order to make the game a good one. One might thus expect the next breakthroughs in adaptive learning systems to come from GBL. We further argue that the advantages to research, offered by the ITS framework, are also potentially beneficial to the way we teach the subject of GBL to our students on game development courses.

Keywords: GBL, ITS, Learner model, Domain model, Teaching GBL

1. Introduction

The first international conference on Intelligent Tutoring Systems (ITS) was in 1988 in Montreal (Gauthier et al., 2000), and it now runs annually. The field of ITS is related to GBL (games based learning), although it emerged directly from Artificial Intelligence in Education (AIED). It aims to make tutoring systems intelligent and adaptive, rather than game-like. There are shared interests however, and such related fields should attend to, and learn from each other.

Here we first describe how the field of ITS emerged from earlier work in cognitive science, that promised a healthy future in learning support systems. Then we introduce the main concepts and terms for this emergent field, to be explored in the rest of the paper.

1.1 A Leading Example of ITS

Emerging from cognitive science in the 1980s, some of the most promising tutoring systems came from researchers at Carnegie Mellon University, led by John Anderson. The specific theory of human cognition deployed was Anderson's ACT-R model (Anderson, 1993, 1996). They reviewed their work, which they called 'cognitive tutors', to find that they usually offered significant achievement gains, in that students learned faster (Anderson et al., 1995). The desired knowledge state was represented in the system as a set of *production rules*, which is a concept from AI that can be thought of as *if-then* rules from situation to action.

Their method was to take an expertly designed curriculum in the domain, and set a sequence of problems for the student to solve. The ideal solution for each problem was then encoded into production rules, and all was packaged into a tutor system with an appropriate interface. All the main parts of the system can be seen as fitting into the more general framework that is described below.

1.2 Concepts to Come From ITS to GBL

As said, the fields of ITS and GBL should be able to learn from each other, by sharing concepts. One such idea that could be taken over is that of the user model (UM). We show below how this and other important concepts in ITS may be brought into GBL with advantage. Let us first introduce some acronyms for these concepts that show how ITS and GBL fields relate to each other in formal terms.

An important framework or architecture for an ITS includes a user model of the learner, which is thus called the LM (learner model). Elementary sorts of information could include the learner's background or entry skill levels, learning goals, or learning styles. Often, it also includes a model for the knowledge domain, that the user is supposed to learn from the system. This DM (domain model) is analogous to a world model in a GBL system (or serious game, SG). It is unusual in GBL to make these models of the learner or domain explicitly *formal*, however: rather the knowledge they would contain and structure is spread implicitly throughout the game software. On the other hand, the field of ITS has often tried to give such formal models a central role, with some success; and this inspires us to explore how such an approach might similarly benefit the field of GBL.

In the following, we describe how formal models of the learner and domain are put into a framework that has organised many systems in the field of ITS, and how successful it has been. The successes have been limited in some ways though, or disappointing; and we look at the publications of the leading ITS conference series to see how the attention has shifted to other research directions, such as recommender systems, that offer other advantages. We suggest some economic and other practical reasons why this shift in the field has occurred.

2. A Modelling Framework for ITS

Since decades, ITS has tried to build upon the use of models, or components, when making systems according to a general framework. In 2010 the conference made a milestone assessment of the history and status of the field’s efforts in this direction (Nkambou et al., 2010). Their classic view of what the chief models should be is pictured in Figure 1, and still organises the field today.

In this view, the models are those components of the software architecture that contain knowledge to support reasoning. They also happen to embody the components or agents of knowledge that teachers already intuitively identify.

The student is depicted at a computer, wanting to learn the subject matter, which in the example is about space travel. She has some notion about what she is going to learn, but it is not yet a full or accurate picture. The system is intended to give her more extensive knowledge about the domain, such as the design of the International Space Station in the example.

The architecture identifies the knowledge that is to be taught to the student as the **domain model** (DM), which adds structure to the information in order to teach it more methodically. In the example a typical notion of structure is illustrated as a *curriculum* of topics or chapters with logical dependencies between them, such as a student will commonly find at the beginning of a textbook. In the example the topics numbered 3 and 4 may be learned in either order, but only after topic number 2.

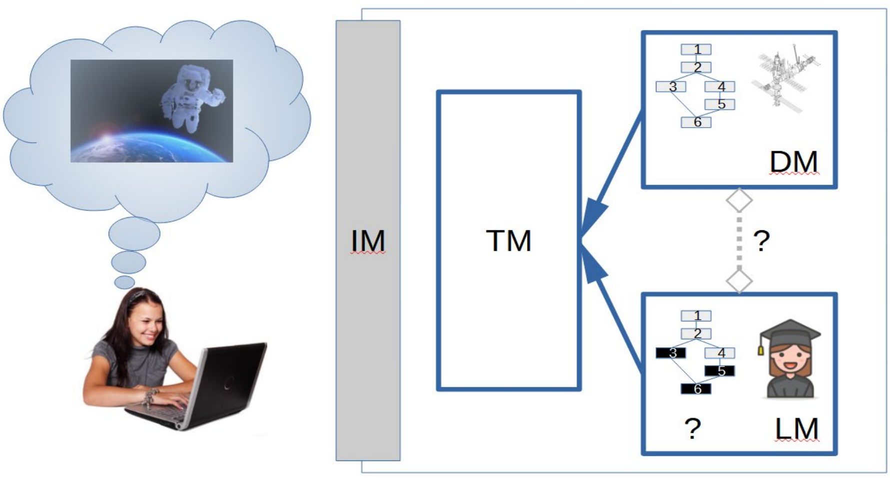

Figure 1: A full ITS framework, showing all the main components of an ITS (intelligent tutoring system). They are the IM (interface model), TM (tutor model), DM (domain model), and LM (learner model)

The student herself is also modelled inside the system, including her current state of knowledge and perhaps other attributes, as the **learner model** (LM), which may also be known as the student model. The picture suggests that she has so far learned some parts of the curriculum (1, 2 and 4), but has not yet gone through the later parts (3, 5 and 6).

Those two models in the architecture are the most obvious and common, and many systems might have one or the other or both of them. In practice, researchers typically have a focus on one main issue, and develop their systems and present their work with an emphasis on the most relevant model in the architecture. The other parts of the architecture are then not identified explicitly, and are often not even identified as separate components in the software. In such cases the functions of the missing modules are either trivial, or still performed in some fashion by the ITS, but only implicitly, as if threaded throughout the code.

A fuller implementation of the architecture would have the other two modules as well, including the **tutor model** (TM) and the **interface model** (IM). These embody the knowledge of how to teach the subject (TM), and how to present each part of the lesson to the student and sense the mental state (IM). In early systems these were combined into a single metaphor of the teacher, to be personified as an artificial agent acting as a person. However, in Figure 1 the tutor is not often represented as a person, but is kept rather abstract. This is because researchers found that students often preferred to see the system as a tool, more like other standard types of computer system. The tutor model thus includes the teaching style or method that a real teacher might deploy, to inform the student or set tasks according to the preferred educational theory.

The framework shown in Figure 1 gives a good general view of how research in ITS can be organised, but it is not definitive. It could be extended in useful ways, and in particular we shall propose an extension of the domain model to world model, for the purposes of GBL.

Before we get to GBL however, let us first see how the framework and its various models have fared over the years. One might expect that the early promise of the approach would lead to accelerated interest, a growing consensus, and greater uptake. That is not what we find in the literature, however.

3. Partial Success of the ITS Framework

It may seem odd that an approach to ITS that was already promising more than thirty years ago has not yet achieved wide recognition, despite its early successes. We suggest some reasons why that may be, below, but first conduct a bibliometric analysis to see how the interest in this modelling approach has developed over time in the field.

We examine the papers from all conferences of ITS, from the year 2000, to compare with later ECGBL conferences (from 2010), to see if the concepts have moved across from one field to the other; from ITS into GBL.

On initial inspection, we noticed that the different sorts of models from the framework (the SM, DM, TM, and IM) are not equally popular. Altogether over the period, there were 182 papers mentioning student models (or learner model or user model). In fact the most common by far are the student and domain models; whereas the other two are comparatively rare, with only one of each occurring in ITS papers as their main focus (Kumar et al., 2014; Raad & Causse, 2002) The first observation we make is thus the uneven distribution of papers about different parts of the framework, and this raises the first question.

One may wonder why most papers only include or refer to one or two models from the whole framework; and can they even be said to fit into the framework if they do not explicitly include all the parts? Firstly, researchers have their own focus of interest, and that focus typically crystallises into the relevant model. Secondly, the framework can still apply even if only one of the models is explicitly involved, as long as we grant that the functions of all the models may still be fulfilled by the system. The function of a tutor model (TM), for example, may still be performed by the system even if it has no such model or code explicitly in its design. This may be because the function is relatively small or trivial; or if it is more significant it may still be left implicit, with the functionality threaded throughout the code.

3.1 Models Have Lost Popularity in ITS

As well as the uneven attention given to different parts of the framework in the field of ITS, there is an interesting change in attention *over time*. We first noticed that papers about learner models appeared to be more common in the early years, of say 2000-2010, with a peak around the year 2006.

To confirm this, we performed a bibliometric analysis by searching for keywords and terms in all the papers: our own collected database for all the ITS papers from the year 2000, and the online ProQuest database for the ECGBL papers from 2010. We used synonyms as alternatives because researchers often use different words but they mean the same general concept. For the concept of the **learner model** (LM) we also searched the terms 'student model' and 'user model'. Table 1 shows the occurrences in ITS papers across all the years to 2022.

Table 1: Occurrence of learner model concept in ITS papers (LM). Conferences were every two years until 2018, and then became annual, so years are combined. Total number of papers per year, published in the proceedings, are in the bottom row.

\ Years Keyword	2000	2002	2004	2006	2008	2010	2012	2014	2016	2018	2019-20	2021-22
Learner	2	4	3	13	8	6	5	3	1	1	3	0
Student	22	13	11	8	6	6	8	10	3	2	3	2
User	6	3	5	1	6	0	1	1	0	0	1	1
SUM LM	33	24	23	25	23	13	16	16	6	3	7	3
Totals	92	117	127	113	120	159	134	110	52	73	113	96

We can see that the synonyms change over time, as one might expect. The term 'user model' was used more in the earlier years, for example, but then got slowly replaced by the more specific terms gaining acceptance, namely 'student model' and 'learner model'. This reflects the establishment of the general framework over time.

Total usage of these terms or concepts, however, first grows quite nicely, but then declines over the years. This can be seen more clearly in the chart of the data, Figure 2, for the concept of learner model (LM).

The chart shows that the concept of learner model was very popular in the year 2000, occurring in about a third of all papers that year. However the concept became less popular over time, dwindling down to less than 5% nowadays.

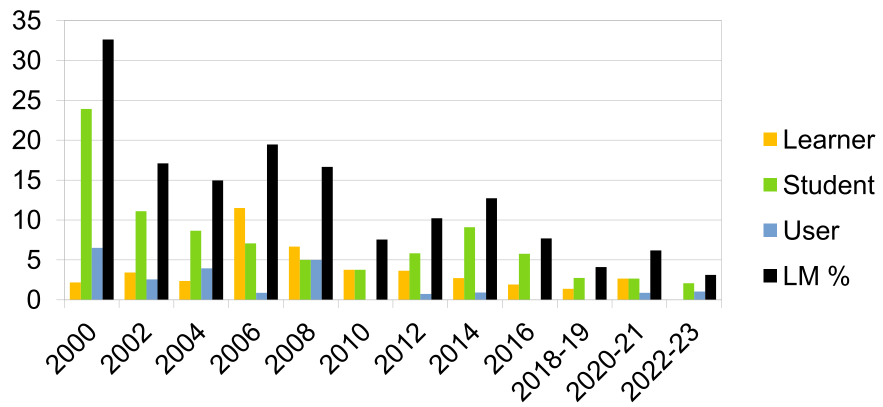

Figure 2: Decline of interest in the concept of learner model (LM) over time. Black bars show the sum for all the alternative keywords. All bars show percentage of papers for that year

The learner model is the most popular model of the framework, and the domain model is next. A similar pattern occurs for the concept of knowledge or domain model (DM), with interest rising to 10% in the year 2008 and then falling away again.

Research interests in a field may change over time for a variety of reasons. One possibility that emerges here is that interests have switched to a new, and more fashionable direction, such as MOOCs (Massive Open Online Course), or 'recommendation systems'. These terms have indeed become more common, generally as well as in the ITS literature; and in fact taken together they make an appearance in 2016 and rise to the levels that DM itself had in 2000-2008. To that extent then we can say that the decline of the ITS framework and its models is partly due to getting supplanted by the latest new fashion.

The decline may signify a deeper malaise, however. After all, MOOCs and recommendation systems do have some advantages, both in feasibility and economics. The MOOC is a course that can easily be scaled up to massive numbers of students with only a technical need of more servers. Recommendation systems are a commercially successful AI technique for recommending products to buyers online, by simple rules based on their previous purchases. They also have a therefore, in that they do not reliably fit well to an individual. Thus we find that MOOCs have a recognised problem of students dropping out, and researchers are still not clear about exactly why that is (Alamri et al., 2019, 2020). In future the ITS approach may come back, as a way to tackle those problems; but first it will need to overcome its own problems that have led to the drop in interest we have found from our analysis above.

3.2 Barriers to ITS

We have demonstrated a drop in research activity, within ITS, that follows the modelling framework set out above. Researchers in the field might not have noticed that drop as such; but some have noted that the field of ITS as a whole has struggled to overcome some barriers (Nye, 2014). Curiously their conclusions are consistent with our own findings from the literature analysis above. They note that barriers related to students have received the most attention, but that other barriers related to teachers and schools have received less. We consider to be consistent with the prevalence of domain and learner models, and the scarcity in the literature of tutor and interface models.

Reviewing the technicalities involved in building the first successful tutoring systems, we can readily understand where some of the difficulties lie. To build a system like those described in section 1.1 above, for example, needs a domain expert to draw up the curriculum, and a specialist to encode the desired knowledge skills into production rules. The specialist needs to be an expert in that type of AI, essentially; an ordinary programmer will not do. The interface model may also need to be modified significantly to suit the school and students. That is a lot of varied expertise that is needed; and even then the task demands time to build and presumably test the system.

4. GBL can use the Framework

Let us first note that gamification has been attractive to the field of ITS because they recognise that lack of student motivation is one key reason for the massive dropout rate in MOOC systems. A few AIEd and ITS researchers therefore add game-like elements into their MOOC courses (Maalejl et al., 2015). However, game design is yet another technical and design skill that would overload the development of such systems more than it is already.

The field of GBL should have better access to game design skills, and so we now explore the suggestion that it could import the framework ideas from ITS to put its own research programme on a more robust footing.

GBL researchers have often been developing along similar or parallel lines. Some have thought in terms of mental models of the learners or players; or have formalised the domain knowledge in order to clarify what they are aiming to teach at different stages of the game-play. However they have also done so without awareness of the parallel and more formally developed work in the field of ITS. This shows that the field of GBL shares the idea that this kind of development lies in a good direction; but also that it has less idea of the virtues of building a shared formalisation framework. We argue that the more formal ITS approach offers several advantages. Many different research projects that use the same formal framework can be more easily compared with each other, to give better measures of progress. Standard benchmarks can be applied to help evaluate research projects, both in performance, and in extent and depth of modelling achieved. The framework and models themselves could, and probably should, also develop. This would reflect the growing understanding of the field as the various micro-theories embedded within are developed, evaluated, and adopted; or discarded in favour of better ones.

As a reviewer of this paper helpfully suggested, we also agree that the field of GBL does need to focus more on cognitive absorption of knowledge, and not merely rely on the supposed attractiveness of games to motivate players. One good way to confront and surmount this issue, which is a kind of obstacle to progress in the field, would be to bring in the ITS approach.

Some papers in the ECGBL conference series have suggested that concepts from ITS would be helpful to make better GBL systems. For example Zapusek & Rugelj (2013) advocate that GBL can learn from ITS, in particular to make serious games more adaptive to the student’s needs. Other authors recognise that there are big technical challenges in making a serious game, and propose an authoring environment to help do this (Tran et al., 2010). That idea arose first in the field of ITS, and they also take over the notion of key components; although they are not exactly the same ones, and they use their own names for them instead of the standard terms that we have mentioned in this paper. While it is not necessary to change names for the concepts, we suggest that it can be a good idea to modify them to suit serious games.

4.1 GBL can Modify the Framework

The concepts of the models that have been identified, for the learner, tutor, domain and interface, are abstract enough to fit many applications. In the case of GBL however, some modifications are clearly needed. The most obvious one would be to change the name of the learner model (LM) to player model (PM). There is one paper from the ECGBL in 2015 that does include a "player model" (Callies et al., 2015).

Another change could typically be useful for the domain model (DM). That is, the domain (or subject material) of the game could be divided into two parts: the world model (WM) and the domain model (again). It is important that the information that the students are supposed to learn should be accurate and correct, and not confused with the fantasy material. This is why we divide the proper domain from the larger world model (WM) than encloses it.

4.2 Possible Advantages to GBL over ITS

The field of ITS has recognised the benefit of the framework, which is why it was proposed in the first place. Their regular researchers have demonstrated the effectiveness of systems based on the framework. However, as we have shown, there are also difficulties which are limiting progress in ITS. We make some observations here to support the notion that GBL research might be able to take the ideas further, due to peculiar advantages of the field.

The nature of interaction with video games is one likely advantage over the more conventional web, and other sorts of interface found in ITS. As we noted in section 1.1, a significant achievement of the first ITS tutor systems was credited to immediate, informative feedback. That happens to be something that most video games excel in!

One economic difficulty with ITS is that it has to make systems based on general infrastructure, like the internet, and associated technologies. If it needs more infrastructure or toolsets then it will have to build them itself.

On the other hand, GBL has access to game technology which comes from the commercial wing of the industry. This includes ecosystems of art assets and other game assets, many of them freely available for educational purposes. It similarly includes sophisticated game engines, platforms, and other auxiliary tools like those for game analytics, which can also be used as *learning analytics*. Such tools give an advantage in making good quality tutoring systems that can run on all platforms.

The games industry has other assets in its game worlds, such as characters for players (avatars) and non-player characters. Much can be done with those, as long as they can be shared affordably. The technology and game worlds may already simulate the regions and parts of the real world, and with high fidelity, which may be suitable as a base for serious games. There are softer resources potentially available as well, including skills of expert game designers, and protocols and other support for play-testing games in their later development stages.

4.3 What the ITS Framework Might Offer to GBL

Along with the prospective benefits, that the field of GBL could offer to the ITS framework, there would also be costs implied in adopting it comprehensively. There should therefore be attractions to the field in return. The framework should be able to help the cause of GBL just as it has already been seen to help ITS, in making tutor systems that teach more effectively. There are also software benefits in making a system out of components, as mentioned above.

We can speculate on further benefits if we consider what is generally the point of an architectural scheme that conceives a system as composed of components such as in the framework. Note that the components are not just any software unit; they are called *models* for good reason. The interface model (IM) for instance is a useful way to isolate all matters of direct interaction with the student (or player), so that the system can more easily be modified for other platforms or contexts. The other models, too, are convenient ways to identify different sources of expertise.

The domain model (DM) has its own domain or subject expert, while the tutor model (TM) naturally embodies the pedagogical philosophy. According to a survey of schoolteachers by Rocha et al (2018), one of the main problems they have with GBL is that they feel the games are *poorly designed*, pedagogically. One way to fix this problem could be to adhere to the framework and include an explicit tutor model (TM), thus allowing the pedagogical method to be modified more easily.

Models in science are often used to predict the outcomes of experiments, and they could also be used to make predictions in the game context. For example, the system might predict how the student or player will perform or react, and adjust the pace of the game accordingly. This allows the game to automatically alter the difficulty level, which is known in games as DDA (dynamic difficulty adjustment).

4.4 Advantages to Education of the Field

So far we mentioned advantages of the ITS approach to GBL as a research field, but it can also help us to educate the next generation of researchers and practitioners. Educational courses in GBL have many things in common, including basic motivation by means of playful interaction, and the use of popular examples of serious games to illustrate what can be achieved. However these courses are often organised along various different lines, according to the taste of the teachers. Just as the research field can be reorganised, such as along the ITS modelling framework suggested here; then in just such a parallel manner can educational courses be similarly organised. In that way, many of the suggested advantages of the framework mentioned above (at the start of this section) would also apply to GBL education.

5. Conclusion

We have conducted a bibliometric analysis of the literature from ITS and ECGBL to track research activity in the important framework concepts of learner and domain models, along with the less popular tutor and interface models. We have shown that the research interest started well and optimistically, but has fallen away since then. This has been partly due to a redirection of attention to other new topics, but also because the framework has run into difficulties and resistance in ITS. Ideas from the framework have been taken up by some researchers in GBL, or at least advocated; but they have not been widely adopted.

We argued that the ITS framework with its models could be used within GBL research to give a coherent structure both to individual systems and serious games, and to the field as a whole. Firstly the field benefits from a more advanced technical base, thanks to sharing infrastructure and other assets with the neighbouring commercial games industry. This gives GBL some advantages over the field of ITS. Secondly, the use of formal models within a serious game enables it effectively to make predictions and thus adapt to the student or player. This could lead to further lines of research like ‘dynamic difficulty adjustment’, and beyond. Thirdly, the models would allow knowledge sharing, to make game development more collaborative; not only by sharing software components, but also the pedagogical theories and the student models that can build up over time, and tested, and sometimes even benchmarked.

There are potential advantages also to using the framework in education, to structure courses in the GBL area. We are currently restructuring one of our own courses in serious games design, to take advantage of the framework. This is done in order to map out the strengths of current research that can be expressed in learner models, as well as to indicate where more research is needed, such as in better design of the cognitive / emotional interactions that should drive learning from the game-play in serious games.

The use of the framework may thus pay for itself in the end, by making individual serious games more adaptive, and intelligent; and also by organising and invigorating the research field of GBL, with stronger methodology.

Acknowledgements

Graphics were in public domain or with creative commons license, from https://www.photosforclass.com/ and https://icon-icons.com/. We thank reviewers for some helpful comments.

References

Alamri, A., Alshehri, M., Cristea, A., Pereira, F. D., Oliveira, E., Shi, L., & Stewart, C. (2019). Predicting MOOCs Dropout Using Only Two Easily Obtainable Features from the First Week’s Activities. In A. Coy, Y. Hayashi, & M. Chang (Eds.), *Intelligent Tutoring Systems* (pp. 163–173). Springer International Publishing. https://doi.org/10.1007/978-3-030-22244-4_20

Alamri, A., Sun, Z., Cristea, A. I., Senthilnathan, G., Shi, L., & Stewart, C. (2020). Is MOOC Learning Different for Dropouts? A Visually-Driven, Multi-granularity Explanatory ML Approach. In V. Kumar & C. Troussas (Eds.), *Intelligent Tutoring Systems* (pp. 353–363). Springer International Publishing. https://doi.org/10.1007/978-3-030-49663-0_42

Aleven, V., McLaren, B. M., Sewall, J., van Velsen, M., Popescu, O., Demi, S., Ringenberg, M., & Koedinger, K. R. (2016). Example-Tracing Tutors: Intelligent Tutor Development for Non-programmers. *International Journal of Artificial Intelligence in Education*, *26*(1), 224–269. https://doi.org/10.1007/s40593-015-0088-2

Anderson, J. R. (1993). *Rules of the mind* (pp. ix, 320). Lawrence Erlbaum Associates, Inc.

Anderson, J. R. (1996). ACT: A simple theory of complex cognition. *American Psychologist*, *51*(4), 355–365. https://doi.org/10.1037/0003-066X.51.4.355

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, Ray. (1995). Cognitive Tutors: Lessons Learned. *Journal of the Learning Sciences*, *4*(2), 167–207. https://doi.org/10.1207/s15327809jls0402_2

Callies, S., Sola, N., Beaudry, E., & Basque, J. (2015). An Empirical Evaluation of a Serious Simulation Game Architecture for Automatic Adaptation. *European Conference on Games Based Learning*, 107–116. https://www.proquest.com/docview/1728409566/abstract/AA512CDF49344D33PQ/1

Gauthier, G., Frasson, C., & VanLehn, K. (Eds.). (2000). *Intelligent Tutoring Systems: 5th International Conference, ITS 2000 Montréal, Canada, June 19–23, 2000 Proceedings* (Vol. 1839). Springer. https://doi.org/10.1007/3-540-45108-0

Kumar, R., Roy, M. E., Roberts, R. B., & Makhoul, J. I. (2014). Towards Automatically Building Tutor Models Using Multiple Behavior Demonstrations. In S. Trausan-Matu, K. E. Boyer, M. Crosby, & K. Panourgia (Eds.), *Intelligent Tutoring Systems* (pp. 535–544). Springer International Publishing. https://doi.org/10.1007/978-3-319-07221-0_68

Maalejl, W., Pernelle, P., Marty, J.-C., & Carron, T. (2015). Integration of Game Based Learning Into a TEL Platform: Application to MOOCs. *European Conference on Games Based Learning*, 321–330. https://www.proquest.com/docview/1728409748/abstract/801372E550B848AEPQ/9

Maciuszek, D., Ruddeck, G., & Martens, A. (2010). Component-Based Development of Educational Games: The Case of the User Interface. *European Conference on Games Based Learning*, 208–XI. https://www.proquest.com/docview/869507901/abstract/A563266226994415PQ/13

Nkambou, R., Bourdeau, J., & Mizoguchi, R. (2010). Introduction: What Are Intelligent Tutoring Systems, and Why This Book? In R. Nkambou, J. Bourdeau, & R. Mizoguchi (Eds.), *Advances in Intelligent Tutoring Systems* (pp. 1–12). Springer. https://doi.org/10.1007/978-3-642-14363-2_1

Nye, B. D. (2014). Barriers to ITS Adoption: A Systematic Mapping Study. In S. Trausan-Matu, K. E. Boyer, M. Crosby, & K. Panourgia (Eds.), *Intelligent Tutoring Systems* (pp. 583–590). Springer International Publishing. https://doi.org/10.1007/978-3-319-07221-0_74

Paquette, L., Lebeau, J.-F., & Mayers, A. (2013). Authoring Problem-Solving ITS with ASTUS: An Interactive Event. In H. C. Lane, K. Yacef, J. Mostow, & P. Pavlik (Eds.), *Artificial Intelligence in Education* (pp. 934–935). Springer. https://doi.org/10.1007/978-3-642-39112-5_151

Raad, H., & Causse, B. (2002). Modelling of an Adaptive Hypermedia System Based on Active Rules. In S. A. Cerri, G. Gouardères, & F. Paraguaçu (Eds.), *Intelligent Tutoring Systems* (pp. 149–157). Springer. https://doi.org/10.1007/3-540-47987-2_19

Rocha, M., Tangney, B., & Dondio, P. (2018). Play and Learn: Teachers' Perceptions About Classroom Video Games. *European Conference on Games Based Learning*, 808-816,XXIII. https://www.proquest.com/docview/2131785733/abstract/801372E550B848AEPQ/11

Tran, C. D., George, S., & Marfisi-Schottman, I. (2010). EDoS: An Authoring Environment for Serious Games Design Based on Three Models. *European Conference on Games Based Learning*, 393–XII. https://www.proquest.com/docview/869507826/abstract/A563266226994415PQ/16

Westera, W., van der Vegt, W., Bahreini, K., Dascalu, M., & van Lankveld, G. (2016). Software Components for Serious Game Development. *European Conference on Games Based Learning*, 765–772. https://www.proquest.com/docview/1859715109/abstract/96E047B827FC40B1PQ/11