Florian Mehm, Stefan Göbel, Ralf Steinmetz: Authoring and Re-Authoring Processes for Educational Adventure Games. In: Academic Conferences Limited: *Proceedings of the 6th European Conference on Games Based Learning*, Academic Conferences Limited, October 2012. ISBN 978-1-908272-69-0.

Authoring and Re-Authoring Processes for Educational Adventure Games

Florian Mehm, Stefan Göbel, Ralf Steinmetz
Multimedia Communications Lab (KOM), Technische Universität Darmstadt

Abstract: Educational Adventure Games attempt to combine positive features of the adventure game genre such as captivating story lines, interesting puzzles and character interactions with educational content. While this approach is promising, analyses of the production processes of typical educational adventure games show that the required interaction between technicians and domain experts introduces delays and higher costs. In order to alleviate this problem, authoring tools unifying these actual processes and practices can be created. This paper describes the application of the Serious Game authoring tool StoryTec to educational adventure games and the insights gained during re-authoring of an existing commercial educational adventure game.
The main contribution of this paper is the demonstration of the elaborated authoring concept by re-authoring an existing educational adventure game, showing the resulting increase in adaptability, extensibility, maintainability as well as the provided level of user support for the groups involved in the creation of such a game. This includes the mapping of previously less structured game content into the structures provided by StoryTec and the application of interaction templates to create the original gameplay found in the game. Thereby, this allows not only extensions to existing games, but to create new educational adventure games with similar designs at lower costs and higher impact at the same time. A focus-group evaluation of the authoring tool carried out with game designers and developers partly involved in the original development of the re-authored game shows the interest in and the feasibility of such an approach.

1. Introduction

Educational Adventure Games are the combination of educational software tools and games from the genre of adventure games. This genre, which is among seven game genres categorized by Gros (Gros, 2007), is exemplified by classic games such as LucasArt's "Monkey Island" series or current games such as "Heavy Rain". It is characterized by the strong focus on a suspenseful or comedic narrative, puzzles to be solved by the players (often by combining items found in the game or by conversing with virtual characters) and the relative lack of action-intensive or time-limited sequences. This combination offers several positive properties for learning: educational content can be transported by means of the strong narrative as well as embedded in the game world and puzzles and players can take their time in assimilating the presented knowledge due to the absence of time limits. From a production perspective, choosing the adventure genre is beneficial: adventure games commonly are not expected to push the limits concerning graphics or effects (which incurs high asset production costs in other game genres), and the interaction in adventure games is often so similar from game to game (with vastly differing content at the same time) that tools for their creation exist (see section 2).
These are reasons why the genre has received the attention of educators for building educational games both during their first era of commercial success – see (Cavallari et al., 1992) – as well as in recent times (Malo and Müsebeck, 2010). An example for a game company producing games for the educational market is the German developer studio BrainGame, exemplified by the Geograficus[1] and Physicus[2] games (in general, Germany can be seen as one of the strongest markets for adventure games worldwide).
While the reasons for choosing the adventure game genre for game-based learning approaches as described above are valid, at the same time creating educational adventure games incurs higher production costs compared to non-educational/purely entertainment-focused games. This is due to the need to include more team members with specialized backgrounds to production teams. Apart from the game designers, artists and programmers required to implement a regular game, domain experts for the domain of the educational game as well as pedagogues for instructional design have to be included. Furthermore, the whole team needs to communicate on issues arising due to this. For example, the game design has to be adapted in order to accommodate the educational content to be taught in the game. Similarly, artists have to be aware of issues concerning the content, such as in a

[1] http://www.geograficus-game.de/
[2] http://www.physicus-return.de/, recently remade for iOS http://itunes.apple.com/de/app//id486422283

historical game where uniforms or villages can't be designed based on fantasy but have to be historically accurate. Processes of coordination such as these increase the production costs and times, often aggravated by different nomenclature in the different specialist groups and different tools used.
The approach we propose for this problem is the introduction of a unified authoring tool, integrating the roles, processes and production stages of game development as described above. This authoring tool allows designers, programmers and domain experts to collaborate during game production. Furthermore, this approach opens the doors to re-authoring of games. Hereby, we use the nomenclature as established by (Rensing et al., 2005), indicating a process where existing content is re-combined to be used in a new scenario. Examples of the commercial viability of re-releasing games are found on the console market, with platforms such as the "Wii Virtual Console" being used by publishers to re-publish vintage computer games on modern platforms.
An additional area where authoring tools can assist during the creation of educational adventure games is the development of adaptive educational games. Such games are able to adapt during runtime to player characteristics such as the level of previous knowledge of the subject matter. This complicates development especially for designers and narrative authors, as they have to provide alternative paths through the game for different adaptations.
The remainder of this paper describes the our approach to authoring and re-authoring of educational games, with a focus on the educational game "Geograficus" by Braingame Publishing, which was re-authored with the authoring tool StoryTec. Section 2 and 3 provide an overview of the state of the art of educational adventure games and authoring tools for this genre. The (re-)authoring process using the StoryTec authoring tool is shown in Section 4, followed by the results of two evaluations of the tool in Section 6. Section 7 finishes with a conclusion and outlook.

2. **Educational Adventure Games**

The idea of using the genre of adventure games for education has a long history in the educational software community. First examples are as old as the genre itself, see for example (Cavallari et al., 1992). The following analysis of the production process of (educational) adventure games is based on the authors' interactions with a commercial educational game developer as well as other accounts from the literature, including (Hodgson et al., 2010) and (Sommeregger and Kellner, 2012).
As with other game genres, the production of an educational adventure game starts with game design. Adventure games are a genre with strong conventions which lead to many of the games conforming to one of a set of paradigmatic games. This can for example be concerning the perspective – 1st or 3rd person; the way in which commands are given to the character – using verbs or symbolic icons or concerning dialogues – using complete utterances in a multiple choice way or only icons representing the theme in which a conversation should continue. Therefore, a major part of the design is the choice among these alternatives. While the basic set of conventions is clearly defined, the remaining content of the game (narrative, characters, backstory, art direction, sound etc.) are very free and are defined in this stage, resulting in a game design document. This includes work by game designers, including the game rules and puzzles, the most important aspect of adventure games. Furthermore, it consists of input by story writers concerning the game's narrative and by concept artists detailing the look and feel of the game as well as the game interface.
For an educational adventure game, domain experts and pedagogues should be included at this stage as well. Domain experts contribute the information about the field to be taught in the game, while pedagogues are concerned with the instructional design, ensuring that different learner types are addressed and that educational content is presented in a beneficial order. This process can also entail the creation of exercise pools or task definitions for the educational aspect of the game. Since the game should integrate both the educational and the play aspects and not have a divide between them, the collaboration between the game developers and the educational experts has to be fostered already at the design stage. For example, the puzzle design can be inspired by the curriculum to be taught and include puzzles which require domain knowledge to solve.
After the design stage has been finished, the game's production commences. This stage is divided into two major strands of work, content production and technical development. The technical basis for the game has to be provided, a task which is carried out by programmers who provide a game engine into which game content and rules are inserted. Thereby, the game's mechanics as described in the game design document have to be realizable in the game engine. Simultaneously, the production of the game's assets (images, 3D models, textures, sounds, GUI elements, etc.) is carried out by artists, based on the game design and the educational requirements. The actual integration of content is usually carried out using programming (scripting) languages and game editors, specialized tools

treated in the next section. Therefore, this task has lower technical requirements than the game engine programming, but still calls for a user trained in the tool and in the used scripting language.
During the game's development and after the content and technical integration has been finished, the quality of the game has to be assured by the QA team. For educational games, this also means testing the educational components of the game and whether the learners in the target group (e.g. concerning age or previous knowledge) are able to efficiently learn with the game.

3. **Authoring Tools for Educational Adventure Games**

The concept of authoring tools as software solutions for editing and composing content for authors stems from the fields of multimedia computing and e-learning. Examples from e-Learning (for an overview, see (Brusilovsky, 2003)) show the general approach of authoring tools: to provide an easy-to-understand and -use interface for content which is then transformed internally by the authoring tool to a platform for publishing, for example as a set of web pages on a web server for e-Learning products.
Authoring tools which can be used for educational adventure games originate from two fields: that of general adventure game editors and that of authoring tools with specialized features for educational adventure games.
The first field is exemplified by a number of tools available both commercially and free. Mentionable are the free Adventure Game Studio (AGS)[3], which has generated a large community of freely available independent adventure games as well as the Visionaire[4] Studio toolset, used both for large recent commercial adventure game releases as well as the educational adventure game Winterfest intended for functional illiterates (Malo and Müsebeck, 2010).
Authoring tools featuring specialized tools for educational game authoring offer the same or similar functionality as those of the group described above, with additional features addressing the needs of an educational game. For example, they can include editors for visualizing or editing the curriculum to be taught in a game, or be linked to a Learning Content Management System to import educational content. In this field, the major system to be seen is the e-Adventure authoring tool (Torrente et al., n.d.), intended for the development of First- and Third-Person adventure games. It provides the possibility to export a game created with the tool to a Learning Management System to be viewed as part of an e-Learning course (Del Blanco et al., 2010) and the possibility to create in-game books to transport knowledge textually.
(Est et al., 2010) describe the SHAI Scenario Editor, a tool abstracting from low-level programming tasks and focusing on providing a graphical interface for defining high-level logic in an educational game-based scenario. This approach is related directly to the ActionSet editor component described in section 4.1, sharing the idea of providing a high-level graphical interface and separating this from the low-level logic of the game.

4. **StoryTec for Educational Adventure Games**

In the following, the use-case of re-authoring the game Geograficus is provided as an example for the authoring process of StoryTec. In order to carry out the re-authoring, the game was first analyzed by means of a thorough playthrough, noting the structure of the game and identifying the necessary interaction templates for implementing the game. An example for such an interaction template would be a hotspot, a clickable area on the game's screen. Another interaction template is that of managing an inventory, a characteristic feature of adventure games (see Figure 3 for the realization in StoryTec).
The use case shown in the remainder of this paper gave the player the task of identifying several gems by their color. This task is embedded in the game in a puzzle surrounding a machine that has to be fixed. The realization of this interaction template is an example for hotspots that can be linked to assets, e.g. to images. Each gem is realized as a hotspot linked to a set of images which are cycled whenever the player clicks the hotspot. The puzzle is solved as soon as the player gets the right combination of gems.

4.1 StoryTec

The basic approach of StoryTec is to integrate the work of the game development roles as described in section 2 into one authoring tool. Both for authoring and re-authoring, this has many advantages.

[3] http://www.adventuregamestudio.co.uk
[4] http://www.visionaire-studio.net/cms/adventure-game-engine.html

Two possibilities for collaboration in game development teams as noted by (Tran and Biddle, 2008) are acknowledged by this concept. It diminishes the centralized role of programmers, thereby removing a possible bottleneck from the project and balancing the responsibilities of the team members. On the other hand, it increases the common vision of the project, since authors all work in a single tool, being able to see the game during each step of the project. The problem of different tools being used and the need to communicate between different groups is also addressed by mapping almost all roles of the game development process into one authoring tool.
The modular authoring tool is composed of several interlinked editors. The first editor to be used commonly by authors is the Story Editor. This editor is used to configure the high-level structure of the game, by breaking down the whole game into a set of scene and defining the transitions between scenes. Figure 1 shows the structure of the use case scene in the Story Editor in the bottom.

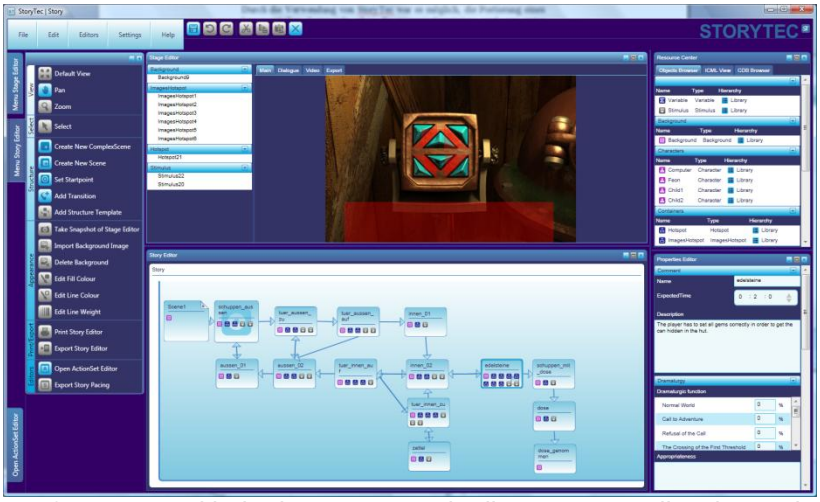

Figure 1: The GUI of StoryTec with the interconnected editors: Stage Editor (upper left), Story Editor (lower left), Objects Browser (upper right) and Property Editor (lower right)

Individual scenes are embellished in the Stage Editor, into which authors can drop objects such as background images, hotspots or videos dragged from the Objects Browser in order to instantiate them. In the case of the use case from Geograficus, the scene in question consisted of a hotspot for navigation purposes (lower highlighted area in the Stage Editor) as well as four hotspots with a cycle of images of gems (upper highlighted area in the Stage Editor).
After scenes have been defined and filled with objects, in a next step properties and metadata of objects are defined. This is carried out via the property editor component. Properties of objects such as hotspots can be the filenames of associated images (e.g. the gems in the use case). For scenes this can be data such as the scene's name or metadata such as the approximate playing time (for control purpose) or parameters for adaptation (see section 5.1).

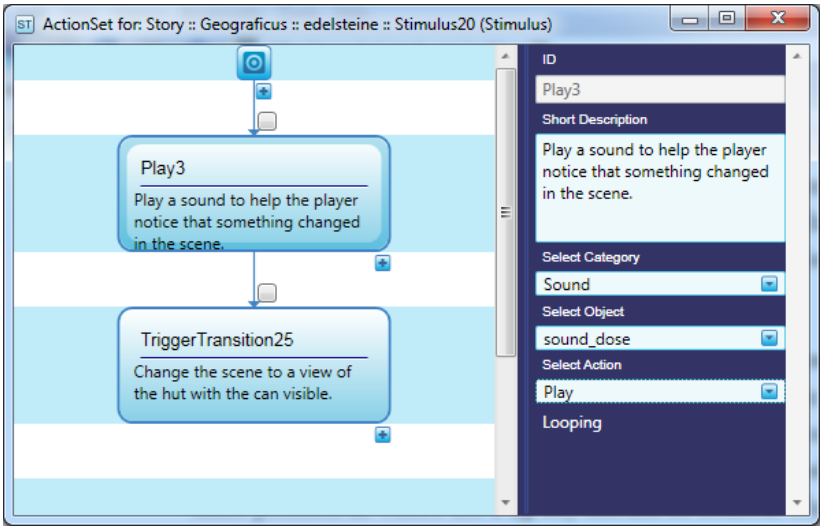

Figure 2: The ActionSet Editor for configuration of control flow during the game

As an alternative to common scripting languages used in game programming, StoryTec relies on the ActionSet Editor (see Figure 2). In this editor, actions to be carried out by the game as the reaction to events triggered by the player (referred to as stimuli) are defined and put into sequences visually.
An example for this are the images hotspots (used in the initial use case to realize the gems), which feature one stimulus (see section **Fehler! Verweisquelle konnte nicht gefunden werden.**), which is triggered whenever a player has set up the correct combination of gems. In this case, the first reaction of the game should be to play a sound for making the player aware of a change in the environment. The subsequent action should be a change of scene to a new image reflecting the change in the scenery. This is realized in the ActionSet Editor by adding two actions in a sequence. Clicking on a sequence allows fine-tuning the parameters of the actions, such as the target object (e.g. a sound file to be played) as well as the exact action to carry out (e.g. “play”) and parameters where necessary (e.g. “play once” vs. “play looping”).
Apart from adding the data for the flow of the game, metadata for game content can be added. These data are subsumed under the term “Narrative Game-Based Learning Objects” (NGLOBs) (Göbel et al., 2010). This encompasses the mapping of scenes to stages of a story model, the definition of learning content including the prerequisite requirements between individual skills/knowledge bits as well as the mapping to types of players with differing play preferences.

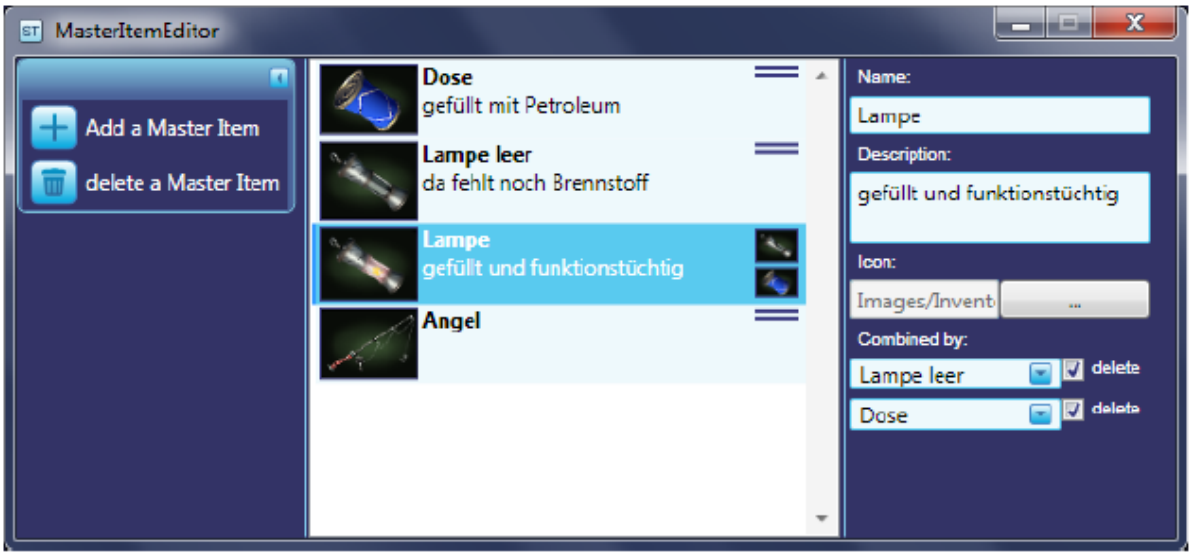

Figure 3: The realization of inventory items in StoryTec

5. Integrated Features

Apart from the advantages of the integrated approach of StoryTec outlined above, several features of StoryTec are positive for authors, both when developing a new game or when re-authoring an existing game.

5.1 Adaptation

Using the concept of NGLOBs (see section 4.1) allows adaptation along the adaptation axes defined by NGLOBs. In the context of a learning adventure, adaptation, especially based on the learning state of the player, is a useful feature. One example could be the increased number of specific tips and background information for players failing several puzzles in a row. Another possibility is the adaptation to play preferences of players using the means of adventure games, for example by offering action-oriented players more action-centered game section e.g. by implementing mini games.

5.2 Introducing new game content

Due to the flexible design and development of games in StoryTec it is possible to add new scenes to an existing game being re-authored. The limiting factor here is the need to create new assets for the added content, e.g. new graphics for new locations. Since individual scenes being added can be tested rapidly (see following subsection), it is possible to test new content by adding it in the form of placeholders (e.g. mock-up level graphics instead of refined art assets) before going ahead with the actual production. In this process, original 3D data are advantageous since they can be re-used for rendering.

5.3 Rapid Prototyping, Testing

The paradigm of rapid prototyping is supported in StoryTec by the player application StoryPlay (Mehm et al., 2010)[5], which allows the simultaneous play of a StoryTec created game and the analysis of context information about the game flow and the internal data of algorithms (e.g. concerning adaptation) as well as the creation of log data, which can be evaluated later on. This allows authors to test new games or extensions to the re-authored games prototypically and decide upon next steps based on the gained insights. This is in line with the recommendations made by (Musil et al., 2010), who argued for an agile approach to game development including iterative development of the game.
StoryPlay is based on the Story Engine introduced in section **Fehler! Verweisquelle konnte nicht gefunden werden.** and is split into a player aspect and an evaluation aspect. Whereas the player aspect allows the interaction with the game, the evaluation aspect offers an overview of gathered data and information. This includes the history of the game until the current point in the play session, an overview of the curriculum of the game and the current state of the player model and the player model built by observing the player's choices in the game.

5.4 Portability

One goal of the development of StoryTec is the deployment of games on various game platforms. The technical realization of this feature on the game platform side was carried out by KTX Software Development[6] and allows the deployment of games on the platforms Nintendo DS, Apple iOS and Browser platforms (see Figure 4). The version of StoryTec available to the public allows deployment on Android devices, HTML 5-compatible browsers and Java-capable desktop devices due to legal constraints.
The portability of games is made possible by the design of StoryTec (see section 4) und leveraging the used xml-based description language ICML as a platform-independent intermediate format. For porting purposes, it is therefore only necessary to port the configurable interaction templates described above onto the target hardware/software. The basic structure and control remain the same across platforms. Necessary steps for adapting the game's content to the peculiarities of the target platform (such as scaling of images to the display resolution) are carried out by a set of tools realizing a pipeline for the creation of platform-specific version.

---

[5] Previously referred to as Bat Cave
[6] http://www.ktx-software.com/

Figure 4: Deployment options of games created with StoryTec

6. **Evaluation**

A first evaluation of StoryTec with other game content was shown in (Mehm et al., 2009). In this section, the results of an evaluation carried out in the context of the Hessen ModellProjekt “StoryTec” are shown, which was based on content of Geograficus.

In order to gain qualitative and quantitative test results, two test methods were used in the evaluation of StoryTec – a usability test in a controlled environment, was well as questionnaire-based test using a questionnaire based on ISO 9241/10. The evaluation was carried out over the course of two days with a total of 26 participants, one of which was female. The mean age of the participants was 25.2 years with a standard deviation of 3.71 years.

During the tests, an adapted form of the “Thinking Aloud” method was used. The participants were given one of three roles – a reader, an performer and an observer. The reader’s task was to read out the steps of the task list aloud. The performer was given control of mouse and keyboard and asked to follow the instructions of the reader. The observer’s role entailed being perceptive and giving additional instructions to the performer in order to understand the software and solve the assigned task.

After completing the assigned tasks, the participants evaluated the tool by means of the questionnaire.

Basic principle	**Mean value**	**Standard deviation**
Suitability to the task	4.74	0.88
Self-description	3.51	0.93
Controllability	5.48	0.77
Conformity with user expectations	4.55	1.06
Error tolerance	3.42	0.80
Suitability for individualization	4.42	0.72
Suitability for learning	5.14	0.78

show the results of the questionnaire grouped to the seven basic principles of the usability standard ISO 9241-10.

A positive tendency in the judgments of the participants concerning the ergonomics of StoryTec can be observed. The lowest mean value is 3.42 on the scale of “Error tolerance”, while the highest mean value is along the scale of “Suitability for learning”.

Table 1: The results of the usability questionnaire

Basic principle	**Mean value**	**Standard deviation**
Suitability to the task	4.74	0.88
Self-description	3.51	0.93
Controllability	5.48	0.77
Conformity with user expectations	4.55	1.06
Error tolerance	3.42	0.80
Suitability for individualization	4.42	0.72
Suitability for learning	5.14	0.78

In order to test if using StoryTec had an influence on the participants' moods, they were asked to rate their current mood on a seven-point psychometric scale supported by graphical icons. To this end, the individual items of the scale were realized as "Smiley" symbols inspired by their use by Kunin (Kunin, 1955) in the measurement of work contentment.
For the pretest measurement, a mean value of 2.72 (standard deviation 0.74) was reached (lower values indicate a positive mood and vice versa). The posttest measurement resulted in a mean value of 2.4 (standard deviation 0.71).. The low difference between pre- and posttest measurement resulted in no statistically significant difference when subjected to a T-Test.
This gives an indication that the use of StoryTec did not have a statistically significant effect on the mood of users. If the ergonomics of the software had been inadequate, the users would have been frustrated, resulting in a negative change in mood.
A second evaluation was carried out in the form of a focus group study with 3 participants (aged 31, 37, 46), all strongly involved in the production of educational games in a commercial game studio. The participants were given similar tasks as during the first evaluation, adapted for only one person. Afterwards, the participants took place in individual guided interviews, including questions about the applicability of StoryTec in their fields of work (including game design, graphics and programming). As a result, it can be noted that the experts were interested to hear about the development of StoryTec and, keeping the prototypical state of the system shown to them, they noted that a full version might be used for game development. In the state it was demonstrated to them, they regarded it as suitable for storyboarding and prototyping.

7. **Conclusion**

We have described both the strengths of using the adventure game genre for game-based learning and the problems arising from typical game development practices. These problems are more pronounced in the case of educational games due to an increase in the need for collaboration and communication.
The basic architecture of StoryTec was presented, intending to unify the software platforms for educational game development by mapping the tasks of almost all involved groups in game teams into various parts of the authoring tool, thereby increasing the potential collaboration between team members and simultaneously structuring the development process. This approach has been demonstrated both for authoring of new games as well as for re-authoring of existing games, allowing upvaluation of the game in both cases by mechanisms such as adding adaptivity, easing portability to diverse hardware platforms or by iterative development. The results of two studies were shown, indicating the usability of StoryTec in the domain of educational adventure games. The viability of StoryTec for educational adventure games has been shown by implementing all required interaction templates for an existing commercial game and authoring it prototypically using representative segments of the game.

References

Del Blanco, A., Torrente, J., Marchiori, E., Martinez-Ortiz, I., Moreno-Ger, P., Fernandez-Manjon, B. (2010). "Easing Assessment of Game-based Learning with e-Adventure and LAMS" *ACM international workshop on Multimedia technologies for distance leaning.* pp. 25-30.
Brusilovsky, P. (2003) "Developing adaptive educational hypermedia systems: From design models to authoring tools." *Authoring Tools for Advanced Technology Learning Environment,* pp. 377-409.
Cavallari, B., Hedburg, J.G., Harper, B. (1992) "Adventure games in education: A review". *Australian Journal of Educational Technology* 8, pp. 172-184.
Est, C.V., Poelman, R., Bidarra, R. (2010) "High-Level Scenario Editing for Serious Games" Richard, P., Braz, J. (Eds.) *GRAPP 2011 - Proceedings of the International Conference on Computer Graphics Theory and Applications*. SciTePress, pp. 339-346.
Gros, B. (2007) "Digital Games in Education: The Design of Games-Based Learning Environments". *Journal of Research on Technology in Education* 40(1), pp. 23-38.
Göbel, S., Wendel, V., Ritter, C., Steinmetz, R. (2010) "Personalized, Adaptive Digital Educational Games using Narrative, Game-based Learning Objects" *Entertainment for Education. Digital Techniques and Systems 5th International Conference on E-learning and Games, Edutainment 2010, Changchun, China, August 16-18, 2010. Proceedings.* Springer, Berlin/Heidelberg, pp. 438-445.
Hodgson, P., Man, D., Leung, J. (2010) "Managing the Development of Digital Educational Games", *DIGITEL 2010 Third IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning*. IEEE, pp. 191-193.

Kunin, T. (1955) "The construction of a new type of attitude measure" *Personnel Psychology* 51, pp. 823-824.
Malo, S., Müsebeck, P. (2010) "Winterfest - An Adventure Game for Basic Education" *Workshop-Proceedings Informatik : 3rd Workshop on Inclusive E-Learning*. Franzbecker, Hildesheim, pp. 61-65.
Mehm, F., Göbel, S., Radke, S., Steinmetz, R. (2009) "Authoring Environment for Story-based Digital Educational Games" Kickmeier-Rust, M.D. (Ed.), *Proceedings of the 1st International Open Workshop on Intelligent Personalization and Adaptation in Digital Educational Games*. pp. 113-124.
Mehm, F., Wendel, V., Göbel, S., Steinmetz, R. (2010) "Bat Cave: A Testing and Evaluation Platform for Digital Educational Games" *Proceedings of the 3rd European Conference on Games Based Learning*. Academic Conferences International.
Musil, J., Schweda, A., Winkler, D., Biffl, S. (2010) "Improving Video Game Development: Facilitating Heterogeneous Team Collaboration Through Flexible Software Processes"Riel, A., O'Connor, R., Tichkiewitch, S., Messnarz, R. *Proc 17th European Systems and Software Process Improvement and Innovation Conference EuroSPI 2010*. Springer, pp. 83-94.
Rensing, C., Bergsträßer, S., Hildebrandt, T., Meyer, M., Zimmermann, B., Faatz, A., Lehmann, L., Steinmetz, R. (2005) "Re-Use and Re-Authoring of Learning Resources - Definitions and Examples" Technical Report (KOM-TR-2005-02). Darmstadt.
Sommeregger, P., Kellner, G. (2012) "Brief Guidelines for Educational Adventure Games Creation (EAGC)" Sugimoto, M., Aleven, V., Chee, Y.S., Fernández-Manjón, B. (Eds.) *Fourth IEEE International Conference On Digital Game And Intelligent Toy Enhanced Learning*. IEEE Computer Society, pp. 120-22.
Torrente, J., Blanco, Á.D., Cañizal, G., Moreno-Ger, Fernandéz-Manjón, B. (2008) "e-Adventure3D: An Open Source Authoring Environment for 3D Adventure Games in Education" *Advances in Computer Entertainment Technology* ACM Press, New York
Tran, M.Q., Biddle, R. (2008) "Collaboration in serious game development" Kapralos, B., Katchabaw, M., Jay, R. (Eds.) *Proceedings of the 2008 Conference on Future Play Research Play Share Future Play 08*. ACM Press, New York, pp. 49-56.