Authoring Serious Games

Florian Mehm

Multimedia Communications Lab (KOM)
Dept. of Electrical Engineering and Information Technology
Technische Universitaet Darmstadt
Rundeturmstr. 10, 64283 Darmstadt, Germany

Florian.Mehm@KOM.tu-darmstadt.de

ABSTRACT

Serious games have large impacts in various fields, as evidenced by the growing scientific and commercial interest in this topic. However, the creation of serious games entails similar processes as those found in game development, which require a large number of specialists from diverse domains. The goal of the research outlined here is the creation of an authoring tool for serious games, which allows non-experts to create games in an environment integrating typical workflows and supporting users in authoring tasks. The challenge of this lies in combining the workflows of E-Learning authoring tools, which hide the complexities of underlying technologies from authors, with the high levels of interactivity and immersion of digital games. To achieve this goal, a model for games will be described, which is manipulated in the authoring tool and executed by the serious game platform developed at the Multimedia Communications Lab. Further research will address processes for supporting authors in various ways, such as rapid prototyping.

1. Introduction

In the past years, the use of games in contexts other than pure entertainment (e.g. in health care, education or cultural heritage) has gained large momentum and today is subsumed under the heading serious games. They combine the inherent properties of games, for example the ease of capturing the attention of players or the learning processes naturally found in well-designed games, with the purposes they are created for.

However, as digital serious games are close kin of regular digital games, their development requires similar processes, which are often driven by experts such as programmers, artists, game designers or sound technicians. In contexts where the necessary expertise and personnel is not available, for example in situations where only one teacher in an educational institution is tasked with the creation of a serious game for a specific course, this is often not feasible due to lack of required skills.

The described situation, in which non-experts are tasked with the creation process, is also encountered in the field of E-Learning, for example in the context of creating web-based trainings (WBTs). In this field, several authoring tools have been created which abstract from underlying technologies such as HTML, JavaScript or Flash, and allow the user to work directly with the contents they plan to integrate. Based on these tools found in the context of E-Learning, a similar authoring tool for serious games can be envisioned. However, the properties of digital games add complexity to this task, as games are typically much more immersive, interactive and non-linear due to their character of often being real-time simulations with tightly interwoven logic between game entities.

As an example, the integration of an image into an E-Learning course and a serious game might be used. In the case of a typical E-Learning authoring tool such as Toolbook[1], the author might drag and drop the image onto a spot inside the current module. This indicates to the system that the image is to be placed at this location. The user can change parameters such as the display size of the image, but conceptually the image is simply inserted into the layout of the module.

On the other hand, in both 2D and 3D computer games, an image might have a multitude of different possible functions. Among them are (animated) sprites, sources for fonts, textures of game objects, or even uses where the image is only indirectly visualized, for example in the form of a height field. Even if the image is only to be shown directly to the user, it is ambiguous how this should be achieved. Variations are for example displaying the image centrally in front of the remaining scene geometry or embedded into the game world (e.g. on a computer screen inside the game)

This example hints at the complexity usually found in digital games and the challenges which have to be faced when creating an authoring system that allows non-experts to handle the integration of content into games. Another problem is that of different levels of interactivity: the interactions in typical E-Learning courses are limited to certain structures, for example multiple choice questions or clozes. An assessment is typically carried out only after the user has finished a whole unit. However, as computer games often incorporate real-time simulations where input is processed constantly, players expect much more immediate feedback to their actions. While it would be possible to build serious games using only exercise types found in classical

---

[1] http://www.sumtotalsystems.com/products/toolbook-elearning-content.html

E-Learning products in the form of templates, this limits the results to a very narrow subset of possible games and excludes the gameplay mechanics that usually differentiate games from classical E-Learning products.

2. Related Work

The work related to the goals of my thesis can be found in the intersection of the areas of E-Learning, game creation tools as well as interactive storytelling. Furthermore, other fields are closely related, such as visual/natural language programming.

E-Learning authoring tools are in place at a multitude of educational institutions as well as companies, and can be used to create educational courses which are presented to users for example via the internet as WBTs. An overview of tools is given in [5].

In the field of tools for the creation of games, commonly used products are often of commercial nature. These tools can be sorted into a spectrum ranging from very complete tools which allow the creation of any kind of game (for example, the Unity game creation suite[2]) to tools linked to one genre or even only one game, which are typically more restrictive. A scientific approach related to the work presented here is the e-Adventure system [10].

Since they are conceptually close to the goals of my research, the related work also includes tools for the creation of interactive storytelling applications. Interactive Storytelling systems provide users with a rendition of a story and allow them to alter the course of the presented story. Of importance to the research outlined here are and will be the findings of the IRIS Network of Excellence [2], which has the study of authoring systems for interactive storytelling as one of its objectives. The Scenejo [9] authoring system is one example from this field.

Visual and natural language programming are approaches to lower the threshold for non-programmers to create programs, by presenting the program logic in more intuitive visual or textual representations which closely resemble natural language. These approaches can be useful when users of an authoring tool have to configure the execution of parts of the authored games. Example systems are the Scratch programming language [8] or the Storytelling Alice system [6].

2.1 Prior Work

The research described here is partially based on the work carried out previously in the Digital Storytelling group of the Computer Graphics Centre in Darmstadt. The major focus of this work was on digital storytelling systems. In this area, several systems and platforms were worked on, among them the INSCAPE platform [1], which influences the current work. Among the components which are used as the basis of my work are a description language for interactive stories (ICML[3]), a story engine which is integrated into other applications as a module and acts as a parser/interpreter for the description language, as well as the story editor component of the INSCAPE project, which assists the author of a digital storytelling application in managing the overall structure of the created story. The description language uses a hierarchical network of scenes which are connected by transitions to model the

structure of a story. Inside scenes, content in the form of various objects can be placed. Interactivity is realized by various concepts, among them that of Stimuli which correspond to outside events (e.g. user input), with which actions are associated.

While the previous work was in some areas conceptually close to games, the shift to authoring serious games requires some adaptation.

3. Proposed Research

The overall aim of my thesis is to research the field of authoring tools for serious games, in order to provide a system using abstractions from underlying formats and engines, adjusted to the properties of digital games.

My proposal for achieving this goal is to first study how a model of objects used in games can be found and described. To this end, a survey of the structures found in many game creation tools is currently being carried out, aiming at identifying the models which exist in games. The derived model will be used in a revision of the description format described in section 2.1.

In a next step, a concept will be developed how the model found can be integrated into the current version of the authoring system (see section 4). A challenge here is the presentation of the possibilities and restrictions imposed by the model to the user. Returning to the example of integrating content such as an image into a serious game, ways of providing the information about possible contexts the image can be used in must be provided to the user. A possibility is providing visual feedback in a revised story editor, similar to the way Scratch [8] visualizes how code pieces can be combined.

In order to shield a user from the complexities of underlying technologies, methods have to be developed to allow the user to work in the authoring tool in an abstract way, while the result is finally executed in a game engine or similar technology. The basis of this can be found in the story engine (see 2.1), which manages the execution of games on a target platform. However, programming complex gameplay and gameplay foundations (e.g. 3D camera motion) without integration of a programming language is either impossible or would lead to highly complex and unintuitive structures. Therefore, these foundations have to be provided by programmers. A methodology that can be used as a paradigm for this process is Component-Based Software Engineering, where components provide services based on agreements formalized as software interfaces. Components in the context of the authoring tool and runtime platform would be gameplay implementations which conform to a certain structure which is presented to the author in the authoring system.

By providing these game components in the authoring tool, a design space is spanned for the author. This space is defined by the possibilities of the used components as well as the modifications the user can make, for example by integrating content where appropriate or by supplying logic elements.

Due to the underlying assumptions of my work, the workflows of various user groups should be supported. For example, a story author is interested in working on the narrative structure of a game's story, while an educator is concerned with the game's instructional design. In some contexts, these workflows have to be handled by one person only. Therefore, the authoring tool should support the different workflows found in the context of serious game creation, and also support users which are not familiar with

---

[2] http://unity3d.com/

[3] INSCAPE Communication Markup Language

a certain workflow. As an example, a user untrained in writing a suspenseful story should be able to create the game's story based on an established story model.

As a specific method of assisting users, I propose to use rapid prototyping in order to enable authors to use an iterative design process (c.f. [11]), where users can generate a working version of a game early during development, test it in a prototyping environment and continue development based on the findings during testing. This process could furthermore be used during the production of the gameplay components, by modeling the components under development in a prototypical fashion (e.g. with reduced functionality).

As a methodology for evaluation of the proposed research, an analysis of whether the implemented system meets the requirements as well as evaluations with users are planned in order to ascertain if the proposed tool can be successfully and efficiently used.

4. StoryTec Authoring Environment

The first year of my work on the topics outlined in this paper was governed by laying the foundations of the authoring system, named "StoryTec" [4,7]. Based on prior work described in section 2.1, a technical framework for the authoring system has been created, which includes editor modules for several different scenarios. The two main scenarios in which authoring is currently possible are in the context of the 80Days [4] project, targeting either the 80Days game platform or a prototyping platform. In the former, players can steer a UFO in an immersive 3D environment realized with the Nebula2 game engine and engage in learning and assessment activities at specified geographical locations. In the prototyping environment, gameplay is rendered in an abstract fashion (e.g. textual descriptions instead of 3D environments). In this demonstrator, the various technologies at work in the 80Days technical platform are visualized, such as the player modeling described in [3].

Furthermore, we are working with a large German publisher and developer of educational games in a project funded by the state of Hesse, in the course of which the StoryTec authoring system is to be extended for the creation of educational game scenarios running on the Apple iPhone, Nintendo DS as well as in a web-based player.

5. Conclusions

In this overview, I have described my plans for research activities leading towards methods and concepts for an authoring system for serious games, which allows non-experts to work on simplified models of games and integrate content into them.

In the first phases of my work, a technical basis for my research has been laid in the form of the StoryTec authoring tool, which has been subject to an early initial evaluation [7]. Furthermore, new concepts such as that of Narrative Game-Based Learning Objects [3] have been integrated into the system. Currently under development is a system for rapid prototyping in the context of the 80Days project.

The next steps will be to research the model of games and game contents to be used in the authoring tool and the integration of this model as a revision of the underlying description format, along with intuitive ways for users to work with this model.

Among the expected contributions during the research are a description of commonly found patterns of objects found in games and game authoring tools, methods for building an authoring tool for such a model, as well as concepts for user support in such a system, including the use of rapid prototyping.

6. References

[1] Balet, O. 2007. INSCAPE An Authoring Platform for Interactive Storytelling. *In Virtual Storytelling. LNCS vol. 4871*. Springer Berlin/Heidelberg, 176-177

[2] Cavazza, M., Donikian, S., Christie, M., Spierling, U., Szilas, N., Vorderer, P., Hartmann, T., Klimmt, C., André, E., Champagnat, R., Petta, P., Olivier, P. 2008. The IRIS Network of Excellence: Integrating Research in Interactive Storytelling. In *ICIDS 2008. LNCS, vol. 5334*. Springer, Heidelberg, 14–19.

[3] Göbel, S., de Carvalho Rodrigues, A., Mehm, F., Steinmetz, R. 2009. Narrative Game-Based Learning Objects for Story-based Digital Educational Games. In *Proceedings of the 1st International Open Workshop on Intelligent Personalization and Adaptation in Digital Educational Games*. 43-53

[4] Göbel, S., Mehm, F., Radke, S., Steinmetz, R. 2009. 80Days: Adaptive Digital Storytelling for Digital Educational Games. In *Proceedings of the 2nd International Workshop on Story-Telling and Educational Games. STEG'09*. CEUR Workshop Proceedings

[5] Horton, W., Horton, K. 2003. *E-learning Tools and Technologies*. Wiley

[6] Kelleher, C. 2006. *Motivating Programming: Using storytelling to make computer programming attractive to middle school girls*. PhD Dissertation, Carnegie Mellon University, School of Computer Science Technical Report CMU-CS-06-171

[7] Mehm, F., Göbel, S., Radke, S., Steinmetz, R. 2009. Authoring Environment for Story-based Digital Educational Games. In *Proceedings of the 1st International Open Workshop on Intelligent Personalization and Adaptation in Digital Educational Games*. 113-124

[8] Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., Kafai, Y. 2009. Scratch: Programming for All. *Communications of the ACM*, November 2009

[9] Spierling, U., Weiß, S., Müller, W. 2006. Towards Accessible Authoring Tools for Interactive Storytelling. In *Technologies for Interactive Digital Storytelling and Entertainment. LNCS, vol. 4326*. Springer, Berlin / Heidelberg, 169-180

[10] Torrente, J., Moreno-Ger, P., Martínez-Ortiz, I., Fernández-Manjón, B. 2009. Integration and Deployment of Educational Games in e-Learning Environments: The Learning Object Model Meets Educational Gaming. In *Educational Technology & Society*, vol. 12, 359–371

[11] Zimmerman, E. 2003. Play as Research: The Iterative Design Process. In *Design Research*, Ed. Laurel, B. MIT Press, Cambridge, 176-184