80Days' 1st International Open Workshop
on

Intelligent Personalization and Adaptation in Digital Educational Games

October 14, 2009
Hotel Weitzer, Graz, Austria

Edited by
Michael D. Kickmeier-Rust

Table of Contents

Program

9.00 – 9.15	**On-site registration and get together**
9.15 – 9.30	**<u>Welcome and Keynote</u>** ***Talking Digital Educational Games*** Michael D. Kickmeier-Rust (University of Graz, Austria)
9.30 – 10.30	**<u>Session I</u>** ***Enhancing Immersion using Personalised Para-social NPCs in Digital Educational Games*** Kevin Koidl (Trinity College Dublin, Ireland) and Daniel Schwarz (Takomat GmbH, Germany) ***The Recipe for Flow Experience*** Kristian Kiili (Tampere University of Technology, Finland)
10.30 – 10.45	**Coffee beak**
10.45 – 12.15	**<u>Session II</u>** ***Rendering Geographic Datasets with 3D Game Engine - Dealing with Compatibility Issues.*** Lorenzo Oleggini, Sam Nova, and Lorenz Hurni (ETH Zurich, Switzerland) ***Teachable Virtual Characters in Educational Game*** Harri Ketamo (Satakunta University of Applied Sciences, Finland) **Narrative Game-based Learning Objects for Story-based Digital Educational Games** Stefan Göbel, André de Carvalho Rodrigues, and Florian Mehm (TU Darmstadt, Germany)
12.15 – 13.30	**Lunch break**
13.30 – 15.30	**<u>Session III</u>** ***Undercover: Non-Invasive, Adaptive Interventions in Educational Games*** Christina M. Steiner, Michael D. Kickmeier-Rust, Elke Mattheiss, and Dietrich Albert (University of Graz, Austria) ***Towards Intelligent Tutoring Systems Based on Computer Role-Playing Games*** Dennis Maciuszek and Alke Martens (University of Rostock, Germany) ***Adapting Contents and Procedures in Educational Video***

	Games with Collaborative Activities Natalia Padilla Zea, Francisco L. Gutiérrez, Nuria Medina Medina, and José Luís González Sánchez (University of Granda, Spain)
	Learner Modeling and Adapted Interaction in Educational Games Kostas Maragos and Maria Grigoriadou (National and Kapodistrian University of Athens, Greece)
15.30 – 15.45	**Coffee break**
15.45 – 17.15	**<u>Session IV</u>** ***The Autopilot - A Personalized Pathfinder in Open Games for Learning*** Franziska Spring-Keller and Helmut Schauer (University of Zurich, Switzerland)
	Authoring Environment for Story-based Digital Educational Games Sequencing story units in Story-based Digital Educational Games. Florian Mehm, Stefan Göbel, and Sabrina Radke (TU Darmstadt, Germany)
	Gender Differences in Perceiving Digital Educational Games: A Mixed-Method Approach Effie Law and Tim Gamble (University of Leicester, UK)
17.15 – 17.30	**<u>Panel discussion and conclusions</u>**

Preface

There is a rapidly growing interest in exploring design and technology of digital educational games (DEGs). Reasons for the present hype over game-based learning are manifold; for example, characteristics of modern computer games enable to transmit knowledge in a very natural and unobtrusive way. Another reason might be that characteristics of computer games amazingly match instructional design principles (e.g., the provision of credible and meaningful contexts for new knowledge). A further reason might be that computer games allow reaching young people, particularly those that are difficult to reach by other means of education. Equally rich as the reasons for the popularity of DEGs are the examples and approaches to game-based learning. The ideas range from utilizing off-the-shelf games to games tailored to specific curricula or age groups, from rich simulation games to augmented reality games, or from game-like enhancements to AAA educational games. More than any other genre of educational technology, computer games do have the potential do make learning and knowledge important and meaningful. In a single word, the key aspect in DEGs is "motivation", the intrinsic motivation to play and therefore to learn. However, more than in any other genre of educational technology, tailoring gaming experience and learning paths to the individual learner is crucial. The intrinsic motivation is heavily determined by gameplay demands, educational demands, personal preferences, and individual factors.

80Days' International Open Workshop on Intelligent Personalization and Adaptation in Digital Educational Games focuses on ideas, approaches, case studies, or technologies, about adapting DEGs to the individual requirements, needs, and preferences of learners. The goal of the workshop is to bring the leading researchers and practitioners in this area together and to initiate a lively discourse.

The workshop is organized by the 80Days Project (www.eightydays.eu), a cutting-edge research initiative in the 7th Framework Programme for research and technological development (FP7). The project's goal is to explore new frontiers in DEGs by combining effective learning with fun and pleasure. Core objectives of the project are intelligent mechanisms for non-invasive knowledge assessment and the establishment of a higher-level adaptation framework that merges educational adaptation and interactive storytelling.

The 80Days Team and I are hoping you have an enjoyable and productive workshop,
Michael D. Kickmeier-Rust

Talking Digital Educational Games

Michael D. Kickmeier-Rust

Department of Psychology, University of Graz, Austria
Brueckenkopfgasse 1 / 6, 8020 Graz
michael.kickmeier@uni-graz.at

Abstract. A common understanding, a common language is an essential requirement for a successful evolution of educational games. A taxonomy that defines terms and, more importantly, sets boundaries and relationships between terms and concepts is deemed to be a necessary step towards a global integration. Moreover, such type of classification serves the avoidance of misunderstandings and myths about games. Therefore, we present a taxonomy of adaptive and adaptable digital educational games, including classifications, hierarchies, and rules. The questions may rise why a framework of educational adaptation is the appropriate context for this kind of taxonomy. The reason is simple; to realize adaptation and personalization it is necessary to know what can be adapted and in which way and in which context, under which regularities, and under which constraints. This is particularly true since "play" is a rather novel factor in educational adaptation and personalization.

The Playing Man

To elucidate the area of digital educational games, we do need a context description of computer / video games per se. Moreover, we have to define what a computer/video game is. Several definitions, taxonomies, and classifications have been proposed.

First, the term game: A popular definition came from the Dutch culture-anthropologist Johan Huizinga in his famous work *Homo ludens* ([1], p.132). He stated, a game "*is an activity which proceeds within certain limits of time and space, in a visible order, according to rules freely accepted, and outside the sphere of necessity or material utility. The play-mood is one of rapture and enthusiasm, and is sacred or festive in accordance with the occasion. A feeling of exaltation and tension accompanies the action, mirth and relaxation follow*". This definition is old, we grant, however, it comprises the most important aspects of games, that is, gratuitousness, enjoyment, rules, and the absence of a purpose. Reversely, this definition is frequently interpreted as if an act has some specific purpose or is conducted by external pressure, it is not a game. This, however, is problematic when it comes to serious games or educational games, which are purposeful and therefore paradox.

Ludwig Wittgenstein [2], emphasized that such simple approach to defining what a game is, fail to comprise the entire concept. Wittgenstein argued that it could not be contained by any single definition; rather games must be considered a "fa*mily resemblance*" of a series of definitions.

A more recent approach came from Chris Crawford [3] who tried to describe the term game along several dimensions such as art, entertainment, play, interaction, etc. This approach may be summarized as an interactive, goal-oriented activity within which players (including virtual characters) can interfere with each other. An attempt to formalize the definition of game on the pillars of challenge, conflict and play came from [4]. These authors argue that the main components (the pillars) are linked together in a subtle way by the representation form (medium), by rules, by the goal definition, and by the absence or presence of opponents (Figure 1).

If we summarize this and many other old and recent definitions, we return to the aspects of Johan Huizinga. A game is characterized by gratuitousness, enjoyment, rules, and the absence of a purpose. A special case is serious games or educational games. In such cases the aspect of the absence of a specific purpose and - in parts - the voluntariness, must be re-considered. An issue we want to point out is the fact that, in an evolutionary and anthropological sense, playing is a purposeful activity, already in animals. The act of playing has the purpose of practicing certain skills, which in turn, closes the circle towards educational games.

Second, computer and video games: The major difference to "traditional" games is the medium with which a computer or video game is transmitted – "a game that is carried out with the help of a computer program" ([4], p. 3). Similarly, we can argue that a video game refers to games that are carried out through game consoles or gaming machines. This distinct might be detailed by referring to input devices ranging from computer keyboards to specific input devices (such as game pads, Nintendo's new Wii controller, or camera controlled inputs). Over the past 40 years, a tremendous diversity of computer game types emerged. This diversity including a broad range of overlap makes it difficult to establish a sound taxonomy of computer games. An early approach came from [3] He differentiated two main categories, the skill-and-action games and strategy games. The skill-and-action games are characterized by real-time play, heavy emphasis on graphics and sound, and use of joysticks or paddles rather than a keyboard. Sub-categories are combat games, maze

Fig. 1. A model of the game concept (Image taken from [4], p.2).

Fig. 2. Example for skill-and-action games according to Crawford (1984): Asteroids, Halo 3, Pac-Man, The Chronicles 2: The Eternal Maze, Fifa Soccer 08, Need for Speed Carbon.

games, sports games, paddle games, and race games (see Figure 2 for examples). The second group, the strategy games, is characterized by an emphasis on cognitive and strategic thinking instead of confrontation or manipulation. Sub-categories are adventure games, role-playing games, strategic war games, games of chance, simulation-type games, and interpersonal games.

A more up-to-date classification of today's computer games is given by [5]. He distinguished six categories (see also Figure 3):

- Action Game
  This kind of games focuses on interactive gameplay and requires fast reflexes and hand-eye-coordination. Among others, sub-types are shooters (ranging from Space Invaders to modern first person shooters), fighting games (e.g., Tekken), survival games (e.g., Silent Hill), or the classic arcade games (e.g., Super Mario).

- Strategy Games
  Strategy games focus on analytical thinking, reasoning, planning. This genre is classified by three sub-categories, turn-based strategy games including minor action components (e.g., Civilization), real-time strategy games (e.g., SimCity), and massively multiplayer online role playing games (e.g., World of Warcraft).

- Adventure Games
  This kind of games focuses on the interactively experiencing narratives. The gameplay itself is, as in strategy games, dominated by cognitive/reasoning aspects. Examples are Lineage or Asheron's Call.

- Simulation Games
  Simulation games are attempting to re-play real or fictitious situations. Generally, such games do not rely on a narrative but put the active (steering) aspects in the foreground. Examples range from Lunar Lander (Atari, 1979) to Microsoft's Flight Simulator. A special sub-category is sports games / sports simulations (e.g., Fifa Soccer 08 or Formula 1 Racing). Another special category is business simulations.
- Puzzle Games
  including matching or constructive puzzle or game play types like Rubik's cube.

- Educational Games
  including simple games for young children, drill-and-practice games.

This taxonomy, however, is quite unsystematic and also incomplete. Certain competitive games genres (competitive sports games) are lacking and so do "analogue" games such as card games, chess, or games of chance. An important systematization of the classification of game genres was made by [6] This approach begins with a classification of games on a 'plane' of ludology, narratology, and degree of reality (the author terms this 'simulation' or 'prosthetic reality'). In a next

Fig. 3. Example games (from top right): Space Invaders, Tekken, Silent Hill, Civilization, SimCity, World of Warcraft, Lineage, Lunar Lander, Flight Simulator.

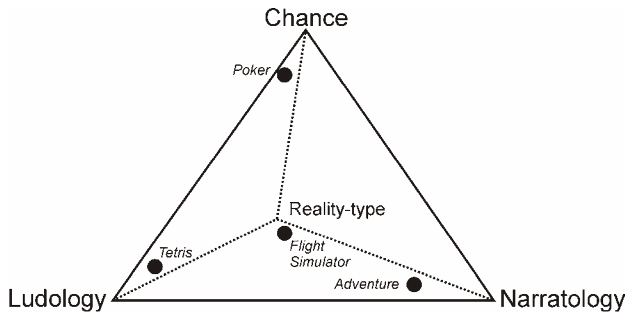


Fig. 4. A three-dimensional classification of games (Lindley, 2003).

step, the model is extended by a 3rd dimension, that of chance (the author terms this 'gambling' or 'decisions about gain and loss'). The model manifests as a three-dimensional pyramid, which allows for classifying game types along its dimensions (Figure 4). Although Lindley's taxonomy offers a systematic approach that covers a wide range of aspects, the purpose aspects is not represented very well. Particularly educational aspects and intentions establish a micro universe of educational game types. The current state-of-the-art can be classified by following categories based on the psycho-pedagogical and technical level of games [7].

- Mini Games for Young Children
  The most common and likely most successful form of educational games are mini games for the preschool age and the primary education level (Figure 5). Generally, those games are based on Flash or Shockwave technology, are distributed through online platforms, and have a child-oriented comic-like design and game-play. In many cases they are associated with characters known from movies, television series, or books. Essentially, the games attempt to teach young children basic knowledge and skills like knowing numbers, letters, simple math, reading, or biology – just to mention a few. Often the game genre is based on trivia, puzzle, memory, or drill and practice (in a positive sense) styles. These games are very successful; they are entertaining and instructional for their target audience.

- Simulation Games
  Simulation games basically pursue a drill and practice approach to certain procedural, strategic, or tactic skills. The instant feedback and risk-free environment invite exploration and experimentation, stimulating curiosity, discovery learning and perseverance [8]. The game character and intensity are varying depending on the context and application. Flight simulators, for example, are used in professional pilot training but almost the same simulations are sold for pure gaming or edutainment purposes (e.g., Microsoft's Flight Simulator). Further examples are military training promoting simulation (games) for practicing specific warfare skills or simulations in medical training (e.g., to enable surgery training in safe

virtual environments). From a technical perspective, simulations and simulation games are on a high level; from an educational perspective, simulations are generally virtual representations of the real world and sparsely implement – if any – sound didactic or pedagogical strategies.

- Off-the-Shelf Games / Moddings
  A third approach to using games for educational purposes is using commercial off-the-shelf games. A prominent example is "Teaching with games", which was a one-year project by Futurelab (www.futurelab.org.uk). Basically, the scope of the project was to research the capabilities of off-the-shelf games for application in schools. The educational objectives of using such games, however, are limited, basically comprising the promotion of collaboration, fostering engagement and motivation, and developing thinking skills. In many cases off-the-shelf games are used as supplementary material or as incentives for learning.
  Modding (a term for modifying software or hardware) is an extension of the off-the-shelf approach. In such cases commercial games are modified using level editors in order to realize certain educational objectives (cf. [9]). An example is the Revolution (Figure 5), a modding of the game Neverwinter Nights, which is supposed to teach and illustrate social aspects of history [10].

- Game-like Enhancements for Learning Material
  A large part of educational games applied successfully in practice is game-like enhancements to digital learning material; [9] classified such games as "tools". Generally, such approaches incorporate sound instructional theories and provide goal-oriented learning situations (LeS). However, in most cases the level of educational objectives, narrative, game-play, and audio-visual realization is limited. In other words, such approach incorporates small games as training for a specific limited set of skills.

- Competitive Educational Games
  What we mean with competitive educational games is games with a primarily educational purpose that – at the same time – can compete with commercial entertainment games as well as with conventional learning environments. Such games are characterized by a convincing narrative and game-play, an appealing audio-visual design, sound instructional design, and clear educational objectives. The most significant characteristic of such games is that learning is embedded in the narrative and the game-play. Among the most advanced learning games are Peacemaker, a commercial game simulation of the Palestine conflict, designed to promote dialog and understanding (www.peacemakergame.com), or NanoMission (Figure 5), a scientifically accurate 3D action adventure teaching nanotechnology through real world practical applications ranging from microelectronics to drug delivery (www.nanomission.org).

Fig. 5. Screenshots of current approaches to educational games (from left to right): A math picture puzzle for young children, the modding Revolution, and the educational game NanoMission.

Based on this brief review of game classifications and taxonomies, the question arises if the existing formalized, systematic taxonomies suffice to provide us with an understanding of the game context and its possibilities, requirements, and difficulties in psycho-pedagogical and motivational-emotional adaptation. The answer must be no. Several authors raised a variety of aspects, however, forgot other aspects. Moreover, a computer-understandable systemic approach is lacking that - at least - includes the aspects of a game's purpose.

To realize this systematic and formal taxonomy, we rely on the aspects raised by previous authors (reviewed here) and propose a collective *hyper-cube taxonomy*. This approach is characterized by an *inner three-dimensional classification core* (Figure 6). This core involves the dimensions:

- **Purpose** – ranging from *synthesizing fun or enjoyment* (or vice versa reducing boredom) to *particular training (learning) purposes*. While the former is straight-forward and mentioned by all existing classifications, the latter aspect is not very common. We want to base this idea on evolutionary theory and animal behavior studies (e.g., that of Konrad Lorenz), that revealed that play is a central learning mechanism in animals.

- **Reality** – ranging from *imitation* of real and fictitious contexts (this includes for example real world simulations but also realistic and credible depictions of fictitious fantasy game worlds) to proving *abstract visualizations* such as in games like *Tetris*.

- **Social Involvement** – ranging from single player games to massively multiplayer games such as *World of Warcraft* or *Second Life*).

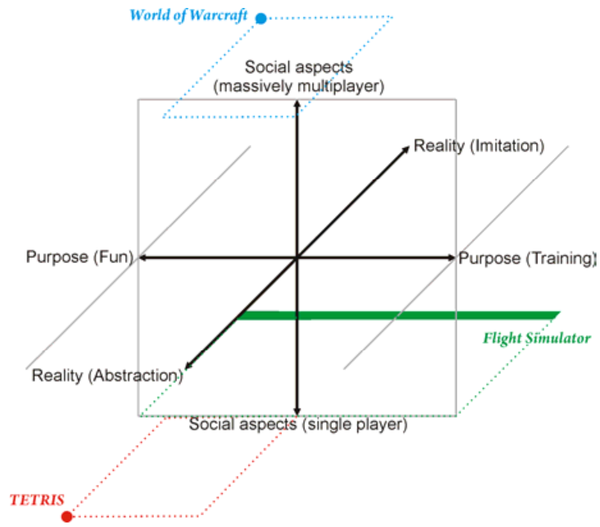


Fig. 6. Inner classification core of the proposed hypercube taxonomy.

This three-dimensional model, in another step, can be extended along the dimensions establishing a four-dimensional hypercube (Figure 7). Therefore, we propose an **activity** layer, ranging from *active game types* (e.g., action games or – even with a physical dimension – the *Nintendo Wii* game play) to *passive game types* (where at the end of this continuum the passive perception of a movie is situated). The continuum not only comprises different general game types but also the amount of passive elements within certain game genres (e.g., an action game can have a high percentage of cut scenes or it might abstain from such movies at all.

Unfortunately, such complex model is not very easy to be overlooked by humans to its full extension. Therefore, the classifying aspects of the hypercube model and the intertwined relationships might be presented with the metaphor of a *property panel* we are all familiar with in several software applications. Such representation is shown in Figure 8. The classifying layers are view, activity, role-play, realism, abstraction, speed, narrative, player, rules, driver, device, motivation, openness, and purpose all located on a bi-polar continuum, which is represented in form o slider controls. All possible combinations of slider settings (which are an infinite number) impose a infinite number of game genres.

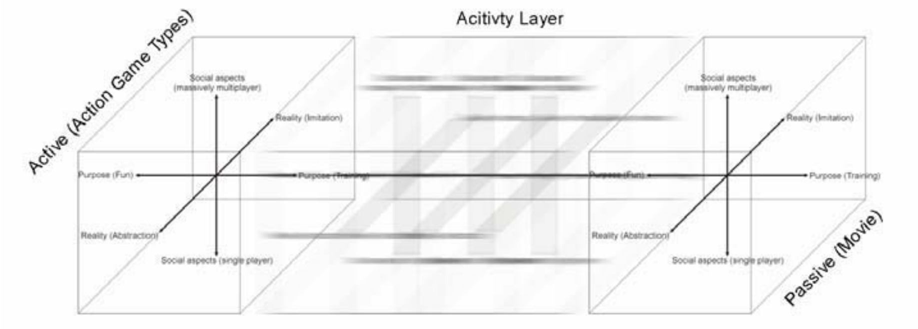


Fig. 7. The hypercube extension of the inner core model. To the left end of the continuum the (even physically active) Nintendo Wii gameplay might be located, at the right and the passive perception of a movie.

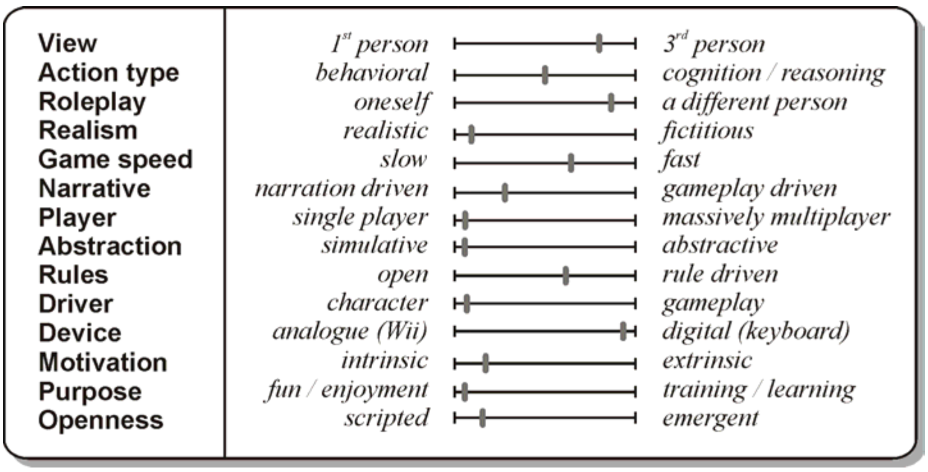


Fig. 8. The hypercube taxonomy model represented in form of a property panel with slider controls. The different layers can be seen as bi-polar continua. This figure shows the famous game series Tomb Raider as an example.

References

1. Huizinga, J. (1938). Homo ludens. Proeve eener bepaling van het spelelement der cultuur. - Homo Ludens: A Study in the Play-Elements in Culture (tr. R.F.C. Hull, 1947). New York: Roy Publishers.
2. Wittgenstein, L. (1953). Philosophical Investigations. G. E. M. Anscombe and R. Rhees (eds.), G. E. M. Anscombe (trans.), Oxford: Blackwell.
3. Crawford, C. (1984). The art of computer game design. Berkeley, CA: McGraw-Hill.

4. Smed, J., & Hakonen, H. (2003). Towards a Definition of a Computer Game. Technical Report, Nr. 553, Turku Centre for Computer Science, Turku, Finland.
5. Schiffler, A. (2006). A heuristic taxonomy of computer games. Retrieved October 12, 2008 from http://www.ferzkopp.net/joomla/content/view/77/15/
6. Lindley, C. A. (2003). Game Taxonomies: A High Level Framework for Game Analysis and Design. Gamasutra, October 3, 2003. Retrieved October 23, 2008 from http://www.gamasutra.com/features/20031003/lindley_01.shtml
7. Kickmeier-Rust, M. D., Peirce, N., Conlan, O., Schwarz, D., Verpoorten, D., & Albert, D. (2007). Immersive Digital Games: The Interfaces for Next-Generation E-Learning? In C. Stephanidis (Ed.), Universal Access in Human-Computer Interaction. Applications and Services (pp. 647-656). Lecture Notes in Computer Science, 4556/2007. Berlin: Springer.
8. Kirriemuir, J. (2002). A survey of the use of computer and video games in classrooms. Internal report for Becta (British Educational Communications and Technology Agency).
9. De Freitas, S. (2006). Learning in immersive worlds. A review of game-based learning. Retrieved August 28, 2007 from http://www.jisc.ac.uk/media/documents/programmes/elearning_innovation/gaming%20report_v3.3.pdf
10. Francis, R. (2006). Revolution: Learning about history through situated role play in a virtual environment. Paper presented at the American Educational Research Association Conference.

The Recipe for Flow Experience

Kristian Kiili

Tampere University of Technology, Pohjoisranta 11A,
P.O. Box 300, FI-28101 Pori, Finland
kristian.kiili@tut.fi

Abstract. Educational games have to be designed properly to incorporate engagement that integrates with educational effectiveness. One foundation to design engagement is flow theory that provides a universal model of enjoyment. The aim of this article is to propose a flow framework that describes the building blocks of flow experience in educational game context. The proposed framework provides a recipe to design engaging game elements that do not conflict with current views on learning. Furthermore, the paper clarifies the relation of flow experience and immersion.

1 Introduction

The ultimate aim of game design is to create appealing experiences to players. Thus, games can be seen only as artifacts that arouse experiences [19]. According to Dewey [6] experience can be described as a continuous interaction between human beings and their environment. Dewey states that the experience is a result of interplay between the present situation and prior experiences. Consequently, players do not have identical playing experiences, but each player's experience is totally unique. This lays a huge challenge to learning game designers. How can we create games that please as many players as possible and are still educationally effective?

The basic elements that comprise every game are mechanics, story, aesthetics, and technology. These are all essential and none of the elements is more important than the others [19]. In case of learning games, the learning objective element needs to be included, which makes the design even more challenging. As Quinn [18] argued, learning-games have to be designed properly to incorporate engagement that integrates with educational effectiveness – the challenge is to find a balance between gameplay and learning objectives. In fact, the designer's task is to balance all the five elements mentioned above in order to create appealing experiences. One foundation to design engagement is flow theory [4, 5]. Flow experience goes beyond the basic game elements because it provides a universal model of enjoyment.

The aim of this article is to propose a recipe for flow experience. The design principles of engagement [13] provide a starting point for this work. In order to be able to understand the background of the factors that contribute to flow experience, the elements that constitute user experience are first distinguished. After that the building blocks of flow experience are described. Finally, the usefulness of flow as a game design framework is considered.

2 User Experience

Design activity has been embraced in an attempt to “design the user experience”. There has been some effort to create models of user experience [e.g. 3, 8, 9, 10], but the theory of user experience is still quite patchy. However, designers of educational artefacts need to understand how users interact with different types of artefacts and how this interaction affects users’ educational experiences.

User experience is usually paralleled to usability. However, usability has been critiqued for not addressing the emotional side of product use enough. Generally, usability aims more at the removal of obstacles than at providing engaging and rewarding experiences. This criticism has ensued from the approach that defines usability from a product-oriented viewpoint, suggesting that usability can be designed into a product. Such an approach considers usability as being ease of use but does not commit to usefulness of the product and quality of use. Thus, more user-oriented and performance-oriented definition is needed. In fact, Bevan and Macleod [1] used the following definition (ISO 9241-11) that takes these aspects into account: “*The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.”* This broad definition of usability enables a shift of focus from features of products also to characteristics and feelings of users. In this paper usability or playability in game context is considered as being one factor among others that affects user experience. Figure 1 shows the author’s macro-level conception about user experience from individualistic point of view. The dimensions are not meant to be understood as overlapping but parallel.

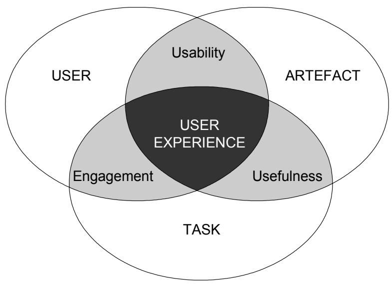


Fig. 1. Elements of user experience

User experience consists of three main elements: the user, an artefact and a task. User experience emerges from the interplay between these elements. The characteristics of a user, such as emotions, values and prior experience, determine how a user perceives an artefact and the task at hand. Usability of an artefact is determined in the interaction between a user and an artefact. Usefulness refers to the design of an artifact containing the right functions required for users to perform their tasks efficiently and to accomplish their goals [15]. Design of an artefact should support a shift from a cognitive artefact-interaction to a fluent one in order to guarantee enough cognitive resources for relevant information processing. Such a shift often means that the use of an artefact is effortless and easily learned [9].

However, not all playing should be effortless. In fact, a learning task should impose a germane cognitive load [21] that is required for knowledge construction. Generally, the way a user perceives a task and an artefact affects user experience. If the task is engaging, the user is willing to use more effort to accomplish the task. Skinner and Belmont's [20] definition of engagement in educational context can be applied to user experience. According to them, engagement refers to the intensity and emotional quality of a user's involvement in initiating and carrying out activities. Engaged users show sustained behavioral and cognitive involvement in activities accompanied by a positive emotional tone. To summarize, good usability, a useful artefact and an engaging task (challenges that the game provides) create prerequisites for a good educational experience. However, it is noteworthy that designers cannot design the subjective experience; only the context arousing the experience may be designed.

3 Building Blocks of Flow Experience

Flow theory provides a meaningful framework to embody new qualities of experience that are relevant for educational purposes. Flow describes a state of complete absorption or engagement in an activity and refers to the optimal experience [5]. During the optimal experience, a person is in a psychological state where he or she is so involved with the goal-driven activity that nothing else seems to matter. An activity that produces such experiences is so pleasant that the person may be willing to do something for its own sake, without being concerned with what he will get out of his action.

The aim of learning game design is to create so interesting experience that it holds player's attention as long and as intensely as possible. Imagine your previous gaming experience when some game totally captured your attention, when the time seemed to fly, when you didn't have any intrusive thoughts during playing, and it felt so rewarding that you wanted to experience it again and again – can you still taste the feeling of flow? Next the ingredients that can be used to crate such engaging experiences are defined. The elements of flow can be divided into three groups: Flow antecedents, flow state, and flow consequences (see Fig. 2).

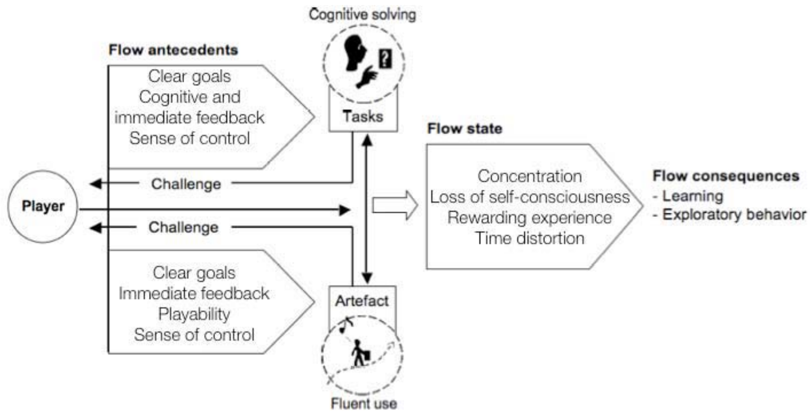


Fig. 2. The flow framework

3.1 Description of Flow Antecedents

The flow antecedents are factors that contribute to the flow state and should be considered in educational game design. Most of the antecedents presented in Figure 2 are consistent with the original flow dimensions [5]. However, playability is a new one. All the antecedents are shortly described below.

When player's goals are clear he can more easily stay focused on the learning tasks. It is good practice to provide a clear main goal at the beginning of the game. The main goal should be divided into sub-goals and provide them at an appropriate pace in order to create feelings of success. If the goals seem too challenging, the probability to experience flow is low. Furthermore, the goals should be related to learning objectives of the game. If the learning objectives are discrete from gameplay the game may fail to produce educationally effective experiences.

The main purpose of the feedback is to inform the player about his performance and progression toward the goals. In the proposed framework, the feedback dimension is divided into immediate feedback and cognitive feedback [2]. Immediate feedback keeps the player focused. If player has to wait long before he can realize what effect his action caused, he will become distracted and loose the focus on the task. Additionally, the delayed feedback may arise interpretation problems and in the worst-case even lead to misconceptions. The cognitive feedback relates to the cognitive problem solving – it is included because it provides the account for learning and cognitive immersion. Cognitive feedback aims to stimulate player to reflect on his experiences and tested solutions in order to further develop his mental models [16] and playing strategies. In other words, it focuses player's attention on information that is relevant for learning objectives. For example, in AnimalClass games [11] player's avatar's (an teachable agent) gestures illustrate the certainty of its knowledge (Fig. 3). Based on the agent's gestures, a player can figure out what his agent knows and what he or she should do next.

Fig. 3. Example of cognitive feedback in AnimalClass game

The playability antecedent is included to replace Csikszentmihalyi's action-awareness merging dimension, which is problematic in the learning game context. This replacement is reasonable, because according to Csikszentmihalyi, all flow inducing activities become spontaneous and automatic, which is not desirable from a learning point of view. In contrast, the principles of experiential and constructive learning approaches emphasize that learning is an active and conscious knowledge construction process. It is noteworthy that reflection is not always a conscious action by a player. However, only when a player consciously processes his experiences can he make active and aware decisions about his playing strategies and thereby form a constructive hypothesis to test. Thus, a distinction between activities related to learning and controlling the game should be made. This means that controlling the game should be spontaneous and automatic, but the educational content related to a player's tasks should be consciously processed and reflected.

Generally, the aim of a learning game is to provide students with challenges that are balanced with their skill level. Furthermore, challenges should be related to the main task so that flow experience is possible. When both the task and the use of the artefact are complex, then the artefact and the task may detract from the player's attention. In fact, bad playability decreases the likelihood of experiencing task-based flow because the player has to sacrifice attention and other cognitive resources to the inappropriate activity. Because the information processing capacity of working memory is limited [17], all possible resources should be available for relevant information processing (the main task) rather than for the use of the game controls. Thus, the aim of the user interface design of games is to support the shift from cognitive interaction to fluent interaction. In an ideal situation, the controls of the game are transparent and allow the player to focus on higher order tasks.

The challenge dimension can be explained with three-channel model of flow [5]. Challenges and skills that are theoretically the most important dimensions of experience are represented on the axes of the model (Fig. 4). The letter P represents a person playing for example snooker. At the beginning (P1), the player has only a little knowledge about snooker and can only perform basic shots. However, the player enjoys the activity (is occasionally in flow) because he feels that the difficulty is just right for his rudimentary skills. While training his basic shots, the player's skills are bound to improve, and he will feel bored (P2) performing such shots. Or he might notice that playing against an opponent is still too hard and he will realize that there are much greater challenges than performing basic shots individually. His poor performance may cause feelings of anxiety (P3).

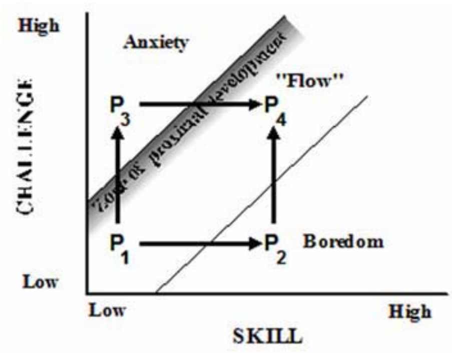


Fig. 4. The extended three-channel model of flow (Modified from [12])

Boredom and anxiety are negative experiences that motivate the player to strive for the flow state. If the player is bored (P2), he has to increase the challenge he is facing. The player can set a more difficult goal that matches his skills. For example, he could play against an appropriate opponent that he can barely win against in order to get back to the flow state (P4). In contrast, if the player feels anxiety (P3), he must increase his skills in order to get back to the flow state (P4). The player could, for example, develop his playing strategy and train to perform safety shots. In general, it can be said that flow emerges in the space between anxiety and boredom. The flow channel can be extended by providing some guidance to the player, or by providing the possibility of solving problems collaboratively. Thus, the zone of proximal development [22] is added to the original model. For example, in the snooker case, the player could ask for help from more proficient players to help him to develop his cue technique and playing strategy.

The model shows that flow is a linear channel where both P1 and P4 represent situations where the player is in the flow state. Although both situations are equally enjoyable, P4 is more complex because the challenges involved and skills required are greater. Neither situations P1 or P4 are stable states, because now and then the player tends to either feel boredom or anxiety, which motivates him to strive for the flow state in order to feel enjoyment again. In conclusion, this dynamic feature explains why flow activities lead to growth and discovery. From the point of view of learning activities, the three-channel model of flow has an important role in that it represents how the process of flow might develop through a single activity. The challenge of the game design is to keep the player in a flow state by increasing the skill level of the game while the skill level of the player increases in order to maximize the impact of them.

Sense of control clearly relates to challenge-skill balance dimension. Csikszentmihalyi [5] has stated that sense of control refers to possibility rather than to actuality of the control. It can be said that a person senses when he can develop sufficient skills to reduce the margin of error close to zero, which makes the experience enjoyable. For example, a rookie snooker player can train hard and dream about perfect skills. However, unconsciously he knows that he cannot ever reach such skill level, but still the illusion, a dream of it, lives and motivates the player to work hard towards his goals, his dreams.

3.2 Description of Flow State

According to Kiili [12], whenever people reflect on their flow experiences, they mention some, and often all, of the following characteristics: concentration, time distortion, rewarding experience, and loss of self-consciousness. In flow a person is totally focused on the activity and is able to forget all unpleasant things. Because flow-inducing activities require complete concentration of attention on the task at hand, there are no cognitive recourses left over for irrelevant information. Thus, self seems to disappear from awareness during flow. In other words, in flow there is no room for self-scrutiny [5]. According to Csikszentmihalyi [5] during the flow experience the sense of time tends to bear little relation to the passage of time as measured by the absolute convention of a clock. Time seems to either pass really fast or the seconds may feel like minutes. Rewarding experience refers to an activity that is done, not with the expectation of some future benefit, but simply because the doing itself is interesting and fun.

3.3 Flow and Immersion

Immersion as a phenomenon resembles flow experience and these phenomena are often confused. Next we try to make a distinction between these phenomena in order to avoid possible interpretation problems that readers may face.

Immersion can be defined as a sensation of being surrounded by a completely other reality taking over all of our attention [7]. Ermi and Mäyrä [7] have divided immersion into three components: sensory, challenge-based and imaginative immersion. Sensory immersion is related to the audiovisual execution of games. Amazing graphics and powerful sounds easily overpower sensory information coming from the real world, shifting a player's attention entirely on the game world and its stimuli. On the other hand, challenge-based immersion concentrates on interaction between the game and the player. It corresponds to Csikszentmihalyi's [5] challenge-skill dimension while it assumes that the feeling of immersion is most powerful when the player can achieve a balance between challenges and abilities. The last component, imaginative immersion enables the player to become absorbed with the stories and the game world, or to identify himself with game characters. Generally, imaginative immersion reflects the possibility of using imagination and enjoying the fantasy of the game.

Although, immersion externally is quite a similar state to flow, it differs from flow in how it captivates a player. In flow a player directs all attention to a certain goal directed activity, whereas immersion means becoming physically or virtually a part of the experience itself. In short, the voluntary direction of attention to relevant content, which is an essential prerequisite for learning, makes the flow theory more interesting from an educational designer's point of view than the immersion based models. However, this does not mean that immersion is considered as an unwanted state, but more like a lower level expression of flow experience, including several important aspects to be considered during game design. Nevertheless, when trying to immerse players we should keep in mind the cognitive constraints of human memory. Thus, the designers should consider for example Mayer's and Moreno's [14] multimedia learning principles when balancing the aesthetics of the games. The meaning of balancing should not be ignored, because too rich game environments tend to arouse incidental processing that may overload a player's mind and disturb learning.

4 Conclusion

In this paper the ingredients that are needed to make a 'Flow Soup' was presented. These proposed flow framework provides a recipe to design engaging game elements that do not conflict with current views on learning. It is important to notice that the flow experience usually occurs when a person's body or mind is stretched to its limits in a voluntary effort to accomplish something difficult and worthwhile. Supporting the flow experience to lead to states of enjoyment does not require educational gaming to be easy and effortless. On the contrary, educational games should stretch a player's mind to its limits in his effort to overcome worthwhile challenges. This nature of flow supports the premise of using flow as a design approach in learning games. However, maybe the most important final result of flow is that flow inducing learning activities are not done with the expectation of some future benefit, but simply because the playing of an educational game itself is the reward. This type of attitude supports the ideology of life-long learning and is priceless goal in education. Although the elements of flow experience were distinguished in this paper, the magic recipe that work in every situation cannot be provided. Designers just have to mix and match the proposed ingredients in appropriate proportion to create their own soup.

References

1. Bevan, N. & Macleod, M. Usability measurement in context. Behaviour and Information Technology, 13, (1994) 132-145.
2. Butler, D. L., & Winne, P. H. Feedback as Self-Regulated Learning: A theoretical synthesis. Review of Educational Research. 65, (1995) 245-281.
3. Buxton, B. Sketching User Experiences: Getting the Design Right and the Right Design . Morgan Kaufmann Publishers, Redmond, Washington (2007)
4. Csikszentmihalyi, M. Beyond boredom and anxiety. Jossey-Bass, San Francisco (1975)
5. Csikszentmihalyi, M. Flow: The Psychology of Optimal Experience. Harper Perennial, New York (1991)
6. Dewey, J. Experience and Education. Simon and Schuster, New York (1938/1997)
7. Ermi, L. & Mäyrä, F.. Fundamental components of the gameplay experience: Analysing immersion. In S. de Castell & J. Jenson (Ed.), Changing views: Worlds in play - selected papers of the 2005 Digital Games Research Association's Second International Conference, (pp. 15-27). Vancouver: Digital Games Research Association & Simon Fraser University.
8. Forlizzi, J. & Battarbee, K. Understanding experience in interactive systems. Proceedings of DIS 2004. Cambridge, MA, 2004, 261-268.

9. Forlizzi, J. & Ford, S. The Building Blocks of Experience: An Early Framework for Interaction Designers. Proceedings of the DIS 2000 seminar. Communications of the ACM, 2000, 419-423.
10. Garret, J.J. The elements of user experience: User-centered design for the Web. New Riders Publishing, USA (2003).
11. Ketamo, H. & Suominen, M. Learning-by-Teaching in Educational Games. In proceedings of Ed-Media 2008. 30.6.–4.7.2008, Vienna, Austria, 2954-2963.
12. Kiili, K. On Educational Game Design: Building Blocks of Flow Experience. Publications of Tampere University of Technology, no. 571. Doctoral thesis, Tampere University of Technology, Finland (2005)
13. Kiili, K. & Lainema, T. Foundation for Measuring Engagement in Educational Games, Journal of Interactive Learning Research, 19 (3), (2008) 469-488.
14. Mayer, R. E., & Moreno, R. Nine ways to reduce cognitive load in multimedia learning. Educational Psychologist, 38, (2003) 43–52.
15. McGrenere, J. & Ho, W. Affordances: Clarifying and evolving a concept. Proceedings of Graphics Interface 2000, Montreal, 2000.
16. Merrienboer, J. J. G. & Kirschner, P. A. Ten steps to complex learning: A systematic approach to four-component instructional design. Lawrence Erlbaum Associates, London (2007)
17. Miller, G. The magical number seven, plus or minus two: some limits on our capacity for processing information. Psychological Review, 63, (1956) 81-97.
18. Quinn, C. Engaging learning: Designing e-learning simulation games. Pfeiffer, San Francisco (2005)
19. Schell, J. (2008). The art of game design: a book of lenses. Burlington, MA: Morgan Kaufmann Publishers.
20. Skinner, E.A. & Belmont, M.J. (1993). Motivation in the classroom: Reciprocal effects of teacher behaviour and student engagement across the school year. Journal of Educational Psychology, 85, 571-581.
21. Sweller, J., Van Merrienboer, J., & Paas, F. (1998). "Cognitive architecture and instructional design". Educational Psychology Review 10: 251–296.
22. Vygotsky, L.S. (1962). Thought and language. Cambridge, MA: The M.I.T. Press.

Rendering Geographic Datasets with 3D Game Engine – Dealing with Compatibility Issues

Lorenzo Oleggini, Samuel Nova and Lorenz Hurni

[1] Institute of Cartography, ETH Zurich, Wolfgang-Pauli-Strasse 15,
CH-8093 Zurich, Switzerland
{Oleggini, Nova, Hurni}@karto.baug.ethz.ch

Abstract. The utilization of 3D computer game engines for the rendering of geographic spatial datasets allows improving certain visualization aspects compared with traditional GIS-visualization. This leads to the possible generation of Geospatial Virtual Environments (GeoVEs) characterized by an increased immersiveness obtained at relatively low costs. Since geographic datasets are generally not meant for being rendered by 3D computer game technology and at the same time 3D rendering engines are not designed for dealing geographic datasets several issues should be tackled when aiming at the integration of Geographic Information System (GIS) data with the 3D rendering world. A geodata rendering approach based on 3D game technology has to deal with issues such as different projections, coordinate systems, spatial resolutions, geographic extents or file formats. This paper aims at illustrating some of the critical points and at presenting the solutions adopted to tackle them within the 80Days project.

1 Introduction

Since the beginning of the 2000s, parallel to the technological advances in computer technology and in particular in the computer gaming industry, several authors has drawn the attention to the possible utilization of 3D gaming engines for an efficient and cost effective realization of GeoVEs. Herwig and Paar [1] recognize 3D game engines as being "versatile and low-cost tools" with the additional advantage of the support of a large user community. Germanchis and Cartwright [2] adopt a 3D game engine for the modeling of an urban area due to the high rendering capabilities and the low cost of this kind of approach. Fritsch and Kada [3] show examples of both in- and outdoor virtual environments created by rendering geographic datasets with computer gaming technology.

Compared with traditional visualization using GIS software this new kind of approach allows real time rendering, increased immersiveness and real 3D modeling (an improvement compared with the 2.5D possibilities of GIS software). Concretely these aspects can be used in order to increase the appealing of GeoVEs. The so developed virtual environments can then be used for instance in computer games generally characterized by low budgets such as Digital Educational Games (DEGs) reducing in this way the costs for the terrain generation. Due to this reason this kind of approach has been chosen for the EU founded research project ‘80Days – Around an inspiring virtual learning world in eighty days’ (80Days; http://www.eightydays.eu). In a first phase freely accessible and utilizable geographic datasets are identified, selected and described and at the same time the generation of a GeoVE using computer game technologies starting from these datasets is tackled. Through this analysis the possibility to produce DEGs cost-effectively should be showed.

This paper describes the principal concerns related with the integration of geographic datasets and 3D gaming technology and presents briefly some tools designed by the authors in order to try to solve the identified issues. In the next section characteristics of geographic datasets are presented and compatibility issues are sketched. In section 3 the tools developed to tackle the identified difficulties are introduced, followed in section 4 by the presentation of some obtained results. Finally in section 5 a conclusion of the presented work is drawn and open issues are discussed.

2 Geographic Datasets Characteristics in Relation to 3D Gaming Technology

Geographic datasets are originally not meant for being rendered by 3D computer game technology but specifically designed for visualization and analysis using typical GIS software. At the same time 3D rendering engines are not designed for dealing with geographic datasets but for handling and visualizing manually modeled 3D data. Therefore a series of compatibility and integration issues are logically expected when aiming at the combination of the GIS with the 3D rendering world. Different projections, coordinate systems and spatial resolutions are some of the aspects to be considered when using geographic datasets outside of a GIS. At the same time, typical 3D technology related concerns such as storage requirements and RAM/memory requisites need to be considered.

2.1 Input Data Related Issues

Geographic datasets are available in various file formats (GeoTiff, DEM...). These specific formats are often designed in order to optimally allow the analysis and interpretation of the data using GIS. The very specific requirements of this kind of software lead often to the definition of proprietary formats and consequently to compatibility issues. From a GIS user's point of view the possibility to support a very large range of formats is an important requirement as data are often collected from different sources and with different techniques. From a point of view of managing these datasets with computer game technology this aspect is not necessarily an advantage. One of the main features of 3D game engines is the high performance at rendering. One of the prerequisite is a file format allowing direct access and utilization of the stored data by the engine. In particular this format should be easily handled and used by the graphic card. As none of the GIS formats is suitable for real-time rendering, the geographic data have to be converted in a more uniform format that can be used directly by the rendering engine.

One of the main characteristics of geographic datasets is their explicit link to real coordinates allowing the spatial referencing of the information. It is therefore possible to carry out investigation and analysis of the data based on the geographic location. At the same time this characteristic simplifies the overlapping and contemporaneous evaluation of data collected from different sources, for different purposes, at different scales. A GIS is able to deal with different coordinate systems and resolution through a series of partly automatic built-in functions and tools. Working on geographic datasets outside of a GIS without considering geographic coordinates will lead in particular to incorrect overlap of data covering different areas. Game rendering engines usually consider the information of the input data starting from a common, previously defined origin (e.g. upper left corner of the dataset). Consequently an overlap only works correctly if the extent, the cartographic projection and the cell dimension of the considered data are perfectly coincident. To solve this problem a method allowing gaming technology to deal with coordinates systems' and cell dimensions' related issues is necessary.

2.2 Storage and Memory Related Issues

Depending on the extent and on the spatial resolution of the covered terrain the required amount of memory might exceed the capacities of an average PC system. In particular this applies if the 3D meshes necessary to render the digital terrain are generated at initialization. This kind of approach is easier to implement but strongly limits the dimensions and the spatial resolution of the covered area. Due to the limited amount of memory a larger area can be covered only by decreasing the spatial resolution of the terrain. Another solution consists in the out-of-core rendering approach [4]. This more dynamic system consists in keeping in memory only parts of the data and streaming further required information dynamically from the storage (hard drive). Special care has to be taken to the delay caused by loading from hard drive. This can be solved using multi-threading so that loading and calculations are done asynchronously to the rendering and the application itself. Most new PC systems have multiple cores/CPUs allowing this more efficient managing of the computation resources.

3 Developed Converting Tools

In order to deal with the aforementioned issues, we developed some tools which are used in 80Days to generate the data for the terrain. The final results obtained with the described tools are presented in Section 4.

3.1 GIS2WORLD

This tool was developed with the purpose of reading GIS data and converting these into a uniformed simple binary format defined in order to satisfy the specific requirements of the project. This format needs to support multiple resolutions of data, and to efficiently deal with geographic regions where no data are available or necessary. Concretely this means that the new format and the tool developed to manage the corresponding data need to be able to store only data for a defined region of interest (generally the area where data are available) without storing at the same time no-data values for the rest of the world. In the concrete case of the 80Days project, textures and height data are stored separately into different file sets using the new defined format. Since GIS2WORLD is only meant for the conversion of the data into a format which is manageable by the 3D game engine the problems related with different coordinate systems and spatial resolution should be solved manually. This means that the user should previously ensure (either by adequately selecting the input data or by modifying them using GIS software) to submit to the tool only data with identical coordinate system and projection. Different spatial resolutions can be managed but needs to be chosen among a pre-defined set of possibilities. The data managed with GIS2WORLD are exported and ready to be visualized by WorldViewer2D.

3.2 WorldViewer2D

WorldViewer2D was first developed as a visual tool to check the data exported by GIS2WORLD but was later expanded to include export of data to be directly used by the terrain rendering engine. As a consequence of the transformation done by GIS2World it is possible to visualize several partially or totally overlapping datasets at once considering their original spatial location. Data are visualized from a top down perspective, the color ramp used for the height data assigns a color to each cell based on its value, textures are shown with the color information of the three bands (red, green and blue) stored using GIS2World. When several resolutions are available it is possible to switch between them. Multiple export areas can be defined allowing for instance the preparation for the game engine of subsets covering specific regions at different resolution.

3.3 PROJECTER.

Similarly to the two previously presented tools, this tool has been developed in order to convert geographic data for the terrain engine. What makes PROJECTER different compared to our previous tools is that it works directly with GIS data at any resolution, any coordinate system and any projection. This way it is not necessary to convert the data beforehand using traditional GIS software. The tool automatically converts all data into one specific format with one specific coordinate system. All areas that are defined for export are defined using geographic longitude and latitude ignoring the resolution of the source data. In case of overlapping data the last defined dataset is used. In a later development phase it will be possible to choose the data with the best or preferred resolution. Even though PROJECTER's exported texture can be viewed directly, height data have to be used within the game engine to be visualized.

4 Results

We adopted the described tools for the generation of freely navigable 3D terrain models as part of the 80Days project. We first applied the approach consisting in the subsequent utilization of GIS2WORLD and WorldViewer2D. The resulting terrain covering a 4000 km by 4000 km wide area of Western and Central Europe has been used in the first demonstrator of the game. A screenshot is showed in Figure 1 illustrating the successful integration of two different geographic datasets: the first one delivering the information about the height of the single points and the second one providing information about the texture.

Fig. 1. Screenshot of the first 80Days demonstrator

In a second phase the approach has been refined and PROJECTER has been developed. An example of the contemporaneous managing of datasets with different spatial resolution is illustrated in Figure 2. This shows the successful replacing of the missing information at higher resolution with a second dataset covering a larger area with a coarser level of details. The revisited approach reduces the limitations of the input data and opens at the same time new possibilities for the development of navigation at different scales as part of a single 3D game virtual environment.

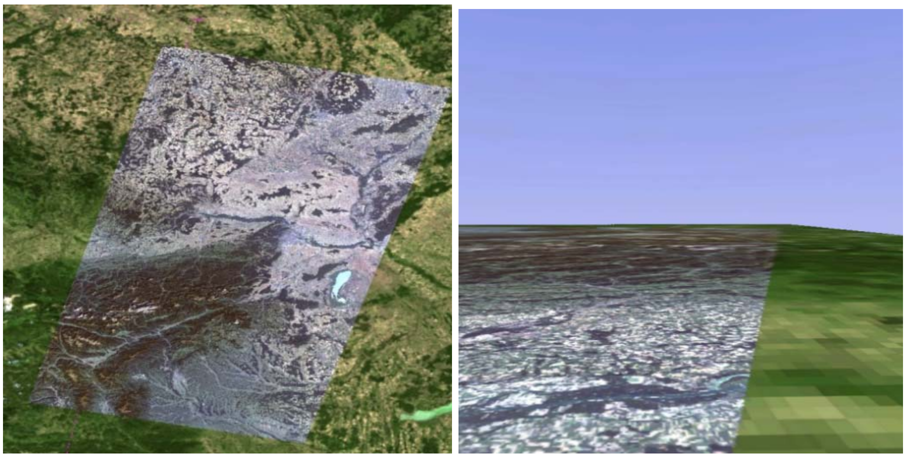


Fig. 2. Overview in 2D of two contemporaneously managed datasets (left) and more detailed, perspective view of the different spatial resolutions (right)

5 Conclusions

In this paper we presented some of the major aspects to be considered when aiming at managing and rendering geographic datasets with 3D gaming technology. At the same time we described three different tools developed as part of the EU-funded research project 80Days allowing us to deal with the issues presented previously. Further improvements of the geographic data management aspects are possible and necessary. Additionally to continuous raster data, discrete geographic information in vector form should also be considered. Parallel to that, problems related to the visual representation of the information contained in the texture images should be tackled. Actually the color range of the images used to project the texture on the terrain is manually corrected with GIS software. This is necessary since generally remotely sensed data do not cover the full range of values displayable by the computer. An automatic display of these values (without applying any stretching) generally results in a dark representation of the scene. Nonetheless the showed results indicate the possibility to concretely use geographic datasets for the generation of digital terrain models with 3D gaming technology. This possibility, requiring few manipulation to the input data, allow the semi-automatic generation of a virtual environment in a cost effective way.

Reference

1. Herwig, A., Paar, P.: Game Engines: Tools for Landscape Visualization and Planning? In: Trends in GIS and Virtualization in Environmental Planning and Design, Proceedings at Anhalt University of Applied Sciences, pp. 162-171 (2002)

2. Germanchis, T., Cartwright, W.: The potential to use games engines and games software to develop interactive, three-dimensional visualisations of geography. In: Proceedings of the 21st International Cartographic Conference, Durban, South Africa, pp. 352–357 (2003)
3. Fritsch, D., Kada, M.: Visualisation Using Game Engines. In: ISPRS 2004: Proceedings of the XXth ISPRS congress: Geo-imagery bridging continents, pp 621-625 (2004)
4. Danovaro, E., Floriani, L. D., Puppo, E., and Samet. H.: Multi-resolution Out-Of-Core modeling of Terrain and Geological Data. In: Proceedings of ACMGIS'05, pp. 143-152 (2005)

Teachable Virtual Characters in Educational Game

Harri Ketamo

Satakunta University of Applied Sciences, Faculty of Business and Culture, Tiedepuisto 3, 28600 Pori, Finland
harri.ketamo@samk.fi

Abstract. The aim of this paper is to introduce Semantic Networks –based game AI and summarize the experiences and findings during design and implementation of the game. The pedagogical idea of Animal Class is to put a learner (player) into the role of a teacher. In Animal Class the player has complete freedom to teach the virtual pet however s/he wants, even wrongly. This possibility of teaching wrongly is a crucial feature in order to enable learning away. The main reason, the AI was built for, is that can we model and produce human-like learning and behavior in a computational way.

1 Introduction

In terms of constructive psychology of learning, people actively construct their own knowledge through interaction with the environment and through reorganization of their mental structures. The key elements in learning are accommodation and assimilation [1][2][3]. Accommodation describes an event when a learner figures out something radically new, which leads to a change in his/her mental conceptual structure. Assimilation describes events when a learner strengthens his/her mental conceptual structure by means of new relations.

The virtual pets in Animal Class learn in a similar way: by adding new concepts to their conceptual structures and strengthening the structure by means of adding new relationships between concepts. This assumed similarity between human learner and machine as learner has been a key element during the design and evaluation of the Animal Class. The traditional goal of AI is to make machines perform cognitive tasks that humans can do, or try to do. In the game industry, the definition of AI is extended so that the most important task of a game's AI is to entertain. It is allowable for game AI to cheat or be 'stupid' in order to achieve the illusion of intelligent behavior [6]. The balancing issue is also challenging within the domains of game development and AI research: It is easy to create a poor or perfect opponent; the challenge is building a reliable and entertaining opponent [4][5].

In Animal Class the intelligence of opponents and game balancing is constructed by game players themselves: the virtual pets are teachable intelligent agents and game mechanics are based upon their behavior. Technically, the game AI in Animal Class is based on a dynamically extensible Semantic neural network [6][7]. Related learnable methods for behavior recording [8][9] and behavior mining [10][11], have been studied and used in the game industry for some time. The biggest difference between methods developed in this study and reviewed methods are in aim: When most of the learnable methods in game AI's are focusing on recording and repeating human behavior in game world, methods and tools in this study aims to model and uncover the causes and consequences behind the playing behavior.

2 Structure of the paper

The aim of this paper is to introduce Semantic Networks –based game AI and summarize the experiences and findings during design and implementation of the game. The game and results received with semantic modeling and/or with game application have been published in several conferences and publications. More detailed scientific results found for example from author's pervious works [12][13][14][15][16].

3 Design of the Game and AI

The pedagogical idea of Animal Class is to put a learner (player) into the role of a teacher. The background of the game is in Learning by Doing, Learning by Teaching and Learning by Programming. In Animal Class the player has complete freedom to teach the virtual pet however s/he wants, even wrongly. This possibility of teaching wrongly is a crucial feature in order to enable learning away.

At the beginning of the game the player got his/her own virtual pet that does not know anything. Its mind is an empty set of concepts and relations. The pet learns inductively: Each teaching phase increases and strengthens the network of concepts. When the pet achieves a semantic network of a certain structure, it can start to conclude. In Animal Class teaching is always based on statements constructed by the player. The virtual pet answers according to its previous knowledge. If there is no previous knowledge, it will guess. The player then tells the pet if the answer is correct or not, and based on this, the pet forms relations between concepts.

A teaching phase consists of a question creation and evaluation –pair. Each teaching phase adds new relations into the conceptual structure. Furthermore, if the concept is not taught before, the new concept is also added into the conceptual structure during the teaching phase. The following example briefly describes the development of conceptual structures in the agent's mind during teaching phases. The understanding of how an agent's conceptual structure develops during playing is important in order to be able to interpret the results of the study. Each teaching phase is recorded in a semantic (conceptual) network within the game AI with one or more 'is (not/option) related to', 'is (not) bigger', 'is (not) equal', etc. relations. The following example is based on is (not) bigger and is (not) equal relations.

At first, the player teaches the relation between 1 and 1/2. The question, created by the player in AnimalClass's classroom is: "Is ½ smaller than 1?" The agent does not have previous knowledge, so it will guess. In case it guesses "true" and the player's evaluation is "Correct." The relation "½ is smaller than 1." is formed in the conceptual structure (Figure 1a). The same would occur in a case where the agent guesses “False” and the player evaluates “Wrong”.

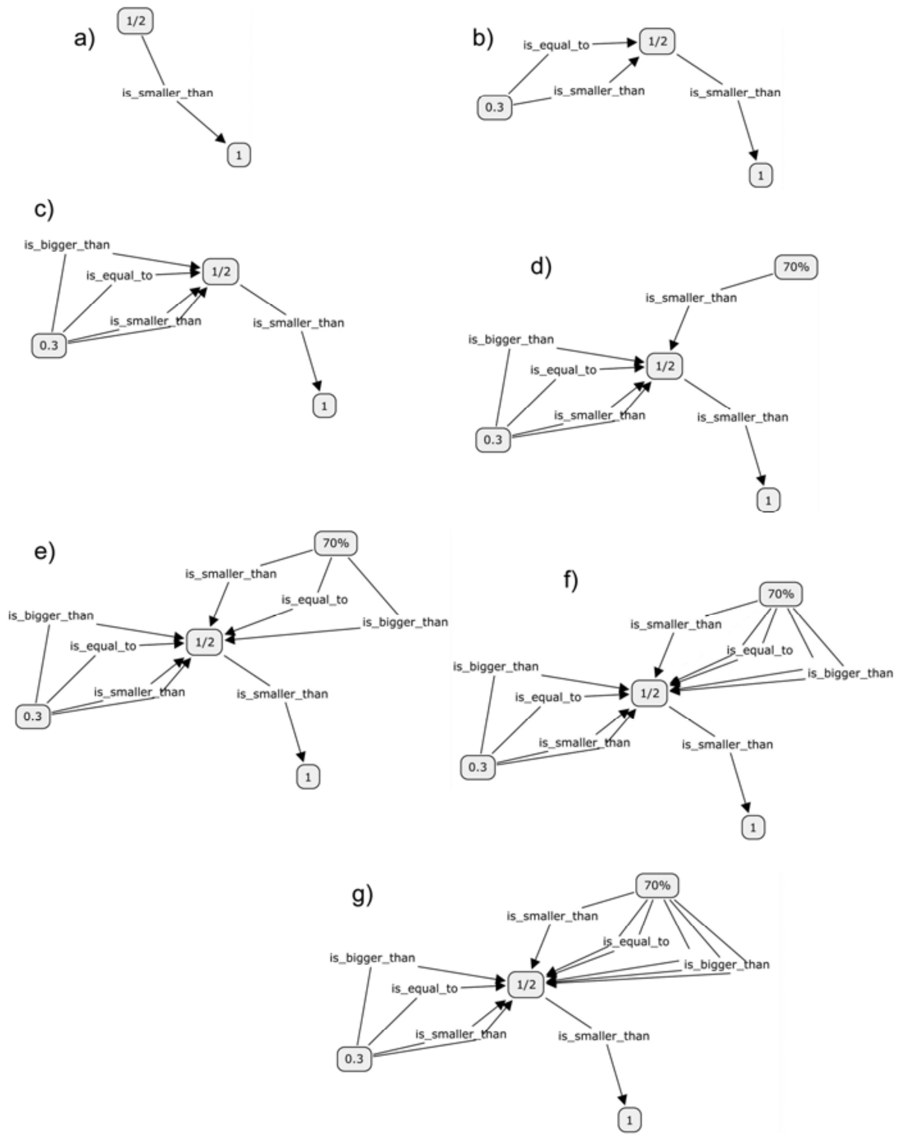


Fig. 1. Semantic network and its development in AnimalClass.

In the second teaching phase, the player teaches a relation between 0.3 and ½, with the question “Is 0.3 bigger than ½?” The player knows that the question is false, but the agent answers (guesses) “True”. So the player evaluates “wrong” and the agent determines that the correct answer is either “0.3 is equal to ½” or “0.3 is smaller than ½”. The conceptual network in the agent’s mind grows by both of these relations (Figure 1b).

In the third teaching phase a player forms a question in another way and asks “is 0.3 equal to ½?”. Again, we know the statement is false. The agent can guess that statement is either “true” according to an “is_equal_to” relation or “false” according to a “is_smaller_than” relation. The agent guesses “false”. When the player evaluates the answer as “correct”, the agent determines that correct answer must be either “0.3 is smaller than ½” or “0.3 is greater than ½”. After adding relations into conceptual structure, the agent knows that the correct answer is “0.3 is smaller than ½” because it is the mode (average) relation (Figure 1c).

In the fourth teaching phase the player asks, “Is 70% smaller than ½?” and on purpose, s/he teaches it the wrong way. The agent guesses that the statement is “true” and the player evaluates the answer as “Correct”, which forms an “is_smaller_than” relation in the conceptual structure (Figure 1d).

In the fifth teaching phase the player starts to correct the conceptual structure. S/He asks again,“ Is 70% smaller than ½?”. According to previous teaching, the agent knows that the answer is “true”. Because the player now knows that it is incorrect answer, the player evaluates it as “incorrect”. In this case the agent determines, that 70% must be equal to ½ or 70% must be greater than ½. After adding relations, the conceptual structure has all the possible comparing statements (Figure 1e) and basically behaves like an empty structure.

In the sixth teaching phase, the player asks for the third time, “Is 70% smaller than ½?”. Because there is no strongest relation, the agent guesses “true”. The player evaluates it again as “incorrect”. Again, the agent determines, that 70% must be equal to ½ or 70% must be greater than ½ and adds those relations to the conceptual structure (Figure 1f).

In the seventh teaching phase, the player decides to change the question to, “Is 70% more than ½?”. The agent guesses “True”, because ‘is_equal’ and ‘is_greater_than’ do contain the same probability. The player confirms that the answer was correct and one more “is_greater_than” relation was added into the conceptual structure (Figure 1g). After that the agent knows that the correct answer is “70% is greater than ½”, because such a set of relations are the strongest.

While playing, the conceptual structure will grow to thousands of relations and a single teaching phase only has a limited effect on the areas of the conceptual structure already taught. Understanding this phenomenon is valuable when trying to correct a wrongly taught part of the conceptual structure. Naturally, wrong teaching could be corrected by teaching the correct structure enough times. The game AI uses all the taught information to back its decisions, and therefore it takes time to override the wrong relations in the agent’s conceptual structure.

An interesting part of teaching is the possibility of teaching wrongly. Sometimes the wrong teaching was not due to low skills: for example at the beginning of geometry game, some pupils tried to teach colors instead of the expected shapes. In order to support reflective thinking, there was a brain icon, (Figures 2 and 3) that describes the quality of learning. If the quality increased, the brains got bigger, and if quality of learning decreased, the size of the brains got smaller. If the overall teaching was wrong, the brains were replaced by a cactus to show the player that he was doing something completely wrong. This kind of wrong teaching could be corrected by teaching correctly long enough to override the wrong learning.

The user interface was designed to be easy to use, but it should give enough freedom to make and evaluate complex expressions. In Figure 2 the player has constructed a question which consists of two triangles and one rectangle. When the question is ready, the player asks the octopus by clicking the 'ask' –button (balloon with three question marks). The octopus answers according to its previous knowledge.

After the octopus has given its answer by pointing out the shape it thinks does not belong in the group, the player should judge the answer: if the answer is correct, the player should click the green 'correct' –button. If the answer is false, the player should click the red 'wrong' –button. If the player notices that he has posed an impossible question or is uncertain, the question can be cancelled by clicking the yellow 'cancel' –button.

Fig.2. Question construction (upper screen) and judging (lower screen) in Animal Class, the Pre-School Geometry Game.

The teaching itself was found to be motivating [12]. Even so, most pupils expected something more than just teaching. Therefore, a quiz challenge called the "Treasure of the Caribbean Pirate" was included into game as a competition between the pets. In the competition the game AI uses the same semantic networks that were taught in the classroom. In the competition a player can challenge his/her friend's octopus to play against him/her. Because all semantic networks are stored in a game server, a player can challenge opponents even if they are not online. The competition (Figure 3) is based on mechanics similar to teaching. The octopus needs to select which of the shapes does not belong in the group. Both octopuses' answer the same questions at the same time according to taught knowledge.

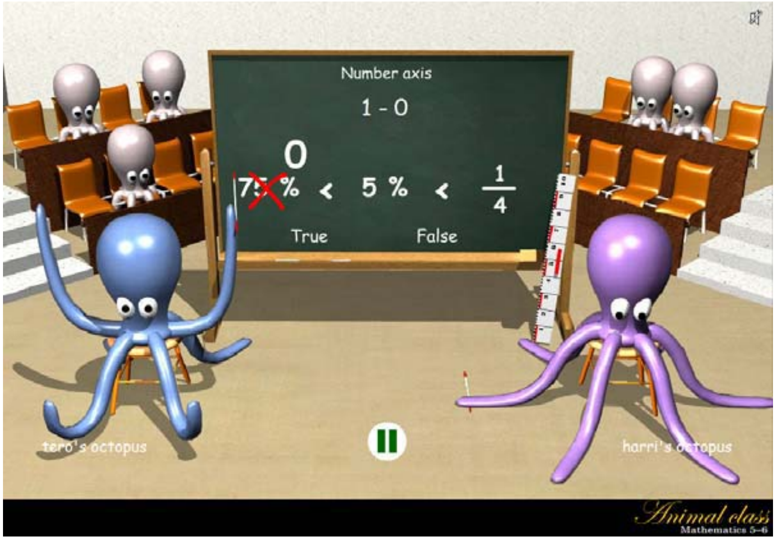


Fig.3. Competition in Animal Class, A 6th Grade Mathematics Game.

The strengths of learning-by-teaching types of games can be summarized in terms of the learning outcome and motivation: More than half of the pupils had good learning outcomes [12][13]. This is caused by meaningful activities that increase learning motivation. One part of this motivation is certainly the possibility to play the game by trial and error -method, in other words, learning by doing. When focusing on conceptual structures built during gameplay, we can see that conceptual changes during gameplay lead to a positive learning outcome.

The weaknesses of learning-by-teaching types of games are related to learning styles. Games are not good for all pupils. Even if pupils like a game, there might still be many better styles of teaching an individual pupil [14][15]. In this study, 40% of pupils did not even benefit from gaming.

When focusing on dependencies between the quality of the taught conceptual structure and measured variables [16], we can find out that the quality of the conceptual structure is strongly related to the post-test score (r=0.457, p=0.009). Because the pre-test and the post-test are dependent (r=0.707, p=0.000) the quality of the conceptual structure is also related to the pre-test score (r=0.370, p=0.037). However, the quality of the conceptual structure was not related to learning outcome (r=0.0118, p=0.949).

5 Conclusions

The aim of the study was to apply machine learning into educational game and evaluate the outcome in context of cognitive psychology of learning. The design and implementation of the game was done in order to support relatively free teaching of a virtual pet in an easy-to-use environment. The game AI was based on a dynamically extensible semantic neural network that enables conceptual learning as well as learning away.

The gameAI is implemented to record the determining rules in order to 1) enable human-like-guesses in the future and 2) to decrease computational costs affected by a network's complexity. In some sense, the game AI in Animal Class learns to learn: the algorithm learns its own inductive bias based on previous experience. Furthermore, the game AI also deals with inductive transfer, or transfer learning: knowledge gained while learning one set of concepts can be applied into new, but related enough, conceptual domain.

A strong relation between quality of taught semantic network during the gameplay and players' knowledge, measured in real life, is an important finding in order to develop new methods for studying conceptual learning in more details. In future studies the attention is paid on combining constructive psychology of learning and reinforcement learning methods in order to build machines that can learn and behave like humans do.

References

1. Baldonado, M., Chang, C.-C.K., Gravano, L., Paepcke, A.: The Stanford Digital Library Metadata Architecture. Int. J. Digit. Libr. 1 (1997) 108–121
2. Bruce, K.B., Cardelli, L., Pierce, B.C.: Comparing Object Encodings. In: Abadi, M., Ito, T. (eds.): Theoretical Aspects of Computer Software. Lecture Notes in Computer Science, Vol. 1281. Springer-Verlag, Berlin Heidelberg New York (1997) 415–438
3. van Leeuwen, J. (ed.): Computer Science Today. Recent Trends and Developments. Lecture Notes in Computer Science, Vol. 1000. Springer-Verlag, Berlin Heidelberg New York (1995)
4. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)

1. Phillips, D.C. How, why, what, when, and where: Perspectives on constructivism in psychology and education. Issues in Education, vol 3, 2, 1997, pp. 151-194.
2. Papert, S. Logo Philosopy and Implementation. Logo Computer Systems Inc, 1999.
3. Vosniadou, S. Conceptual change approach and its re-framing. In Vosniadou, S., Baltas, A. & Vamvakoussi, X., (Eds), Re-Framing the Conceptual Change Approach in Learning and Instruction. Oxford: Elsevier Press, 2007, pp. 1-15.
4. Scott, B. The Illusion of Intelligence. In Rabin, S. (ed.) AI Game Programming Wisdom. Massachusetts: Charles River Media, 2002, pp. 16-20.
5. Liden, L. Artificial Stupidity: The Art of Intentional Mistakes. In Rabin, S. (ed.) AI Game Programming Wisdom II. Massachusetts: Charles River Media, 2003, pp. 41-49.
6. Reye, J. Student Modelling based on Belief Networks. International Journal of Artificial Intelligence in Education, vol 14,2004, pp. 63-96.
7. Kim, K.M., Hong, J.H. & Cho, S.B. A semantic Bayesian network approach to retrieving information with intelligent conversational agents. Information Processing & Management, Vol 43,1, 2007, pp. 225-236
8. Bowling, M., Furnkranz, J., Graepel, T. & Musick, R. Machine learning and games. Machine Learning, vol 63, 2006, pp. 211-215.

9. Houlette, R. Player Modeling for Adaptive Games. In Rabin, S. (ed.) AI Game Programming Wisdom II. Massachusetts: Charles River Media, 2003, pp. 557-567.
10. Mukkamala, S., Xu, D. & Sung, A.H. Intrusion Detection Based on Behavior Mining and Machine Learning Techniques. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2006, pp 619-628.
11. Kuo Y.H., Huang, Y.M., Chen, J.N. & Jeng, Y.L. Extended Real-Time Learning Behavior Mining. In proceedings of Fifth IEEE International Conference on Advanced Learning Technologies (ICALT'05), 2005, pp 440-441.
12. Ketamo, H. & Suominen, M. Ways to support reflective thinking in educational games: Gaming Strategies and Learning. In proceeding of Online Educa Berlin 2007, 28-30.11.2007.
13. Kiili, K. & Ketamo, H. Exploring the Learning Mechanism in Educational Games. Journal of Computing and Information Technology, vol. 15(4), 2007, pp. 319-324.
14. Ketamo, H. & Suominen, M. Learning-by-Teaching in Educational Games. In proceedings of Ed-Media 2008. 30.6.–4.7.2008, Vienna, Austria., 2008, pp. 2954-2963.
15. Kiili, K., Ketamo, H., & Lainema, T. Reflective thinking in games: triggers and constrains. In proceedings of European Conference on Game Based Learning, 24-26.11.2007, Paisley, UK,, 2007, pp. 169-176.
16. Ketamo, H. Semantic networks -based teachable agents in an educational game. WSEAS Transactions on Computers, vol 8(4), 2009, pp. 641-650.

Narrative Game-based Learning Objects for Story-based Digital Educational Games

Stefan Göbel[1], André de Carvalho Rodrigues[2] and Florian Mehm[1]

[1]Multimedia Communication Labs – KOM, TU Darmstadt
Merckstr. 25, 64283 Darmstadt, Germany
{stefan.goebel, florian.mehm}@kom.tu-darmstadt.de
[2]carvalho_rodrigues@stud.tu-darmstadt.de

Abstract. This paper introduces the concept of Narrative Game-based Learning Objects (NGLOB) as basis for the creation and control of Story-based Digital Educational Games (DEG). In the context of the European research project 80Days, section 1 describes major aims and challenges of Story-based DEG and motivates for the introduction and use of NGLOB. In the main part, section 2 provides an overview of the conceptualization of NGLOB and its formalization within ICML as XML-based format for Story-based applications and DEG. In section 3 the theoretical results are set in context to the 80Days approach. Finally, the main results are summarized and further research activities are outlined.

Keywords: Digital Educational Games, Interactive Digital Storytelling, Technology-Enhanced Learning, Narrative Game-based Learning Objects.

1 Motivation

The overall aim of the European research project 80Days[1] situated in the field of Technology-enhanced Learning is to combine adaptive learning, storytelling and gaming technology in order to build intelligent, adaptive and exciting learning environments in the form of Story-based DEG.

Fig. 1. Technology-enh. Learning

Hereby, the major challenge is to harmonize the different characteristics and objectives of the storytelling, learning and gaming approaches. In simple words, key aspects and aims of these approaches might be summarized as:

- Storytelling: Use of stories as instrument for suspenseful knowledge transfer.

[1] 80Days – Around an Inspiring Virtual Learning World in Eighty Days. EU, FP7, IST, STREP, Challenge 4.1.2 Technology-enhanced Learning. www.eightydays.eu

Keywords include dramaturgy, suspense or emotion and immersion.

- Gaming: Provision of a playful learning environment. As slogan *'Learning by playing'* might serve. Fun, motivation, exploration and interaction are dominant.
- Learning: Most relevant is the knowledge transfer. Emphasis is set on assessment, learning success and effectiveness or methodic-didactic aspects. Mechanisms to motivate and engage users are welcome.

In major parts, these aspects and characteristics are complementary, for instance both storytelling and gaming concepts are used to increase the motivation of users in digital educational games. On the other hand –especially with respect to a technical implementation and integration of the concepts– the so-called "narrative paradox" [15] indicates a conflict between storytelling (narratology; linear, non-interactive, plot-based approach) and gaming (ludology; interaction, non-linear gaming approach). Consequently, the role of the author is different: Whereby in storytelling approaches the author has full control over the DEG run-time scenario (everything is pre-scripted, no interaction, no choice to choose among learning paths → "aura of the author"), in gaming approaches the player more or less has full control and decides how a Story-based DEG continues (the story evolves during gameplay → "emergent narrative"[14, 16).

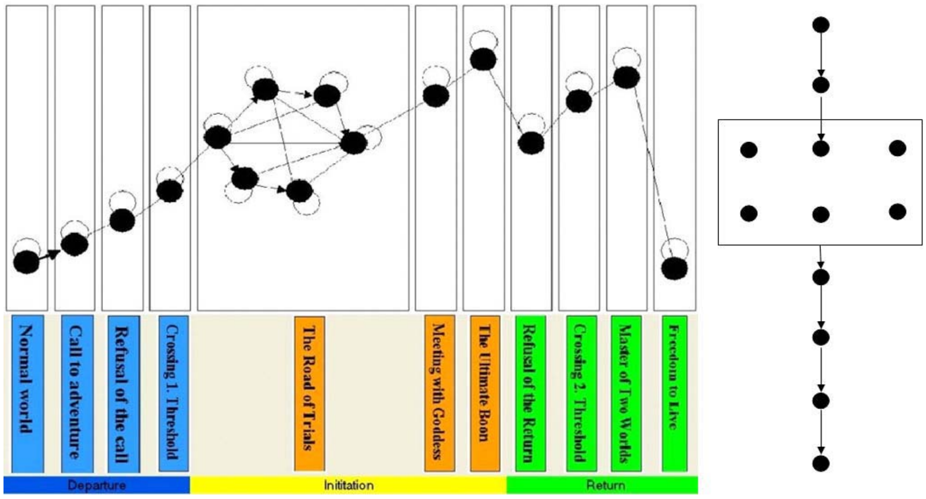


Fig. 2. Hero's Journey story model (left) – linear and modular story units (right).

From a Storytelling perspective the narrative paradox and the question how a Story-based DEG continues during run-time (→ macro adaptation, sequencing) has been at core of research within the first period of the 80Days project and previous work of the authors of this paper [12, 13]. As outcome of comprehensive analytic studies and conceptual work, a compromise between plot-based Storytelling and non-linear, interactive gaming approaches has been identified [6, 7] and builds the conceptual basis for the overall story structure of 80Days´ demonstrator 1: The hero's journey [3, 17] which is well-proven especially in the domain of adventure games, is used as underlying (in major parts linear) story model in order to 'guarantee' a suspenseful story. Hereby, the middle part with the dramatic step 'The Road of Trials' is very flexible and provides the possibility to integrate as many story units (→ so-called 'micro missions' in 80Days – similar to levels in a DEG) as the author wishes. Then, during run-time, the story and sequence of micro missions evolves based on the player interaction respectively a mixture of a) pre-defined rules by the author, b) player/learner model and c) learning, gaming and storytelling context.

For that, the main questions are '*How to formalise that multi-faceted context?*' and '*What makes a narrative, game-based learning object (NGLOB)?*'

The following sections of this paper concentrate on the conceptualisation of NGLOB and its use to create and control Story-based DEG. The definition and consideration of player/learner/user models as well as (micro) adaptation using NGLOB according to learner/user/player preferences are not part of this paper, but topic of ongoing research within the 80Days approach at the Serious Gaming group at TU Darmstadt.

2 Conceptualization of Narrative Game-based Learning Objects

In a first step, based on examples of the game design document (GDD) for the first demonstrator of the 80Days approach, the formalization of narrative, gaming and learning objects is briefly discussed, before an integrated model and its technical implementation in form of an XML-based schema is introduced.

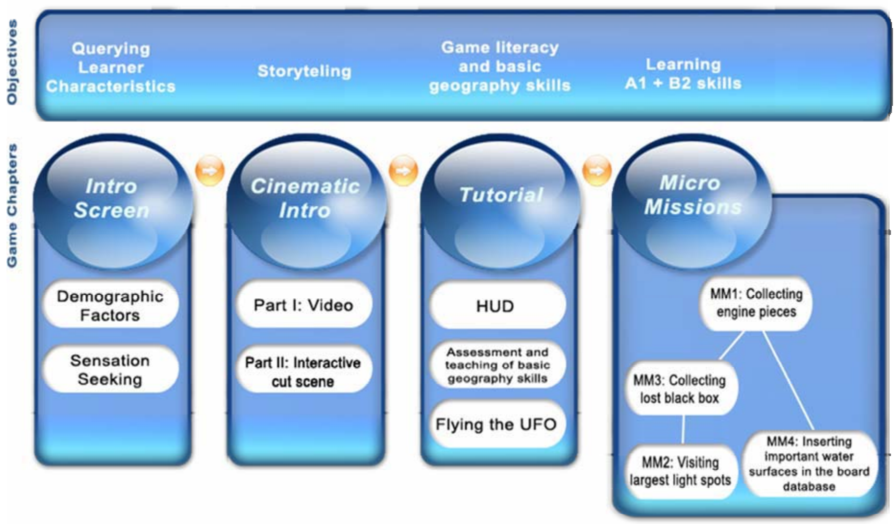


Fig. 3. Story Structure of 80Days´ demonstrator 1. Source: Uni Graz.

More or less following the story model of the Hero's Journey described above the overall story structure of 80Days consists of story units: An *Intro Screen* with preassessment to categorize player/users, a *Cinematic Intro* to initiate the story – corresponding to the first part ‘departure’ of the Hero’s Journey, an *Interactive Tutorial* introducing the topic and providing information about the gameplay – might be understood as ‘Crossing 1. Threshold’ and the micro missions *MM1* to *MM4* representing different game levels or quests – referring to ‘The Road of Trials’ in the middle part of the Hero’s Journey. Each micro mission or the Cinematic Intro provide an unique ID and short description/synopsis, optionally a visual representation (e.g. a sketch) and it is split into storytelling (StS), learning (LeS) and gaming/gameplay situations (GpS) on a narrower level.

Game Chapter	**Cinematric Intro**				
Situation	*Short description*	*Picture*	*Function for Story*	*Function for Learning*	*Function for Gameplay*
CI_scene 1	We see the original NASA film footage of the Apollo 8 mission..		Create the beginning of a mystery story; set the mood and frame of the story..	This is a visual metaphor for our philosophy for teaching Geography..	--

Table 1. Extract out of the GDD for the Cinematic Intro of 80Days´ demonstrator 1.

The situations themselves vary in its emphasis on either one specific context (e.g. a pure story-driven situation without any gameplay and minor/nor learning effects, see situation CI_scene1 in table 1) or a combination of contexts (e.g. LeS 1.3 as game-based learning situation, see table 2).

Game Chapter	**Learning Unit 1 (LU1): B2 Skills European Capitals & Countries**			
Situation	*Short description*	*Function for Story*	*Function for Learning*	*Function for Gameplay*
LeS 1.1 Pre-test of existing knowledge	Alien asks boy: “You know what cities are these?” The boy now can link illuminated spots and city names on a desk. ..	Now, Mr. Jackanapes has to struggle a first time to keep up his blarney of being an all-knowing earthling.	Reflection on and pre-test of existing knowledge without immediate feedback. ..	Introduction: Game play mode “Global view/Map desk” in simplified 2D view.
LeS 1.3 Position of cities without known names	The gamer can fly above Europe in the UFO and the 2D night map in the HUD gives him his precise position and supports him in deciding to which city (light spot) he wants to fly next.		The player can freely explore Europe while having the learning goals on a map in front of him (cities shown as light spots).	To verify the cities´ names the player has to fly there and to stay paused above them (logging).

Table 2. Extract out of the GDD for Learning Unit 1 of 80Days´ demonstrator 1.

In summary, that kind of style for a GDD might be useful for authors and serve as compact storyboard, but from a technical point of view the problem is the lack of metric, quantifiable information being necessary to be interpreted and processed by computer systems – e.g. in order to determine whether a situation is appropriate to fulfill a specific dramaturgic step within the story model of the Hero’s Journey or not.

Therefore, a major aspect during the conceptualization of NGLOB has been set on the integration of measurable, quantitative and qualitative elements and annotations of narrative, gaming and learning contexts.

2.1 Narrative, Learning and Gaming Context

From a Storytelling perspective, narrative objects (NOB) represent the smallest, atomic units of Story-based DEG. In 80Days, NOB are implemented as storytelling situations and technically implemented as cut scenes (without any interaction) or as speech acts and additional actions of virtual characters (animations, gestures, etc.).

For the formalization of NOB and narrative contexts, the idea is –as far as applicable– to map and annotate NOB corresponding to the steps and dramaturgic functions of underlying story models such as the Hero’s Journey.

With respect to learning issues, formal models for learning objects (LOB) and its use within courseware (Computer and Web-based Training) have been researched for a long time in the field of E-Learning (for instance with a focus on data storage [11] and standards are already available: Whilst SCORM[2] focuses on the definition of the overall structure of online courses and provides mechanisms for sequencing LOB (part ‘Sequencing and Navigation’, available since version SCORM 2004), LOM provides a set of metadata elements for the description of LOB and to facilitate search, evaluation, acquisition, and use of LOB.

For the conceptualisation and formalization of LOB in the context of Story-based DEG it is not the aim to rebuild the LOM standard. Contrary, the approach is to use existing learning resources and to reference on it within the content section of the overall story format ICML (see section 2.2). Additionally, the idea is to formalize the learning and learner context and to provide machine-readable information about associated and prerequisite skills of a LOB respectively learning situation based on the Competency-based Knowledge Space Theory (CbKST) provided by Albert and Lukas [1]. Thus, for sequencing purposes –presumed an open, modular, emergent (narrative) environment is available without hardcoded transitions as in pure linear approaches (see section 1)– it is possible to decide whether a learning situation is appropriate for a specific learner (the learner does have the prerequisite skills) or not (the learner would be overstrained).

Opposite to the learning context, unfortunately, there are no well-known, elaborated definitions, standards or formats for the gaming context and gaming objects (GOB). Though, different generic descriptions of games á la ‘What makes a good game?’ indicate relevant characteristics and criteria: Apart from graphic and audio/sound, especially gameplay, interaction concepts and a good story decide about success or failure of a game. With respect to the formalization of NGLOB and its use

---

[2] SCORM: Sharable Content Object Reference Model, www.adlnet.org/Technologies/scorm

in educational games, the interaction concept in form of interaction templates (e.g. drag-and-drop, multiple-choice and puzzle templates in classic courseware or an explorative 3D environment such as a flight mission in 80Days) provides useful attributive information.

Second, similar to the learning context, the idea is to build a correlation between gaming situations/GOB and the users (i.e. players in the gaming context) and underlying player models. Hence, all gaming situations are set into context with player types and annotated with appropriateness factors.

2.2 Model and Technical Implementation of NGLOB

In sum, the model for a NGLOB (≈ situation in 80Days) is built by a composition of context information resulting in a triple vector C_N x C_G x C_L.

The narrative context C_N provides a list of tuples (storymodelStep, appropriatenessFactor) whereby the storymodelStep is encoded by the initials of a storymodel (for instance ‘HJ’ for Hero’s Journey) plus a number for the step/part of that model. The parameter appropriatenessFactor is normalized in the range [0..1].

The gaming context C_G primarily tackles the appropriateness of individual GOB and gaming situations for different players and player types. Analogue to the narrative context C_N, the gaming context C_G also provides a list of tuples (playerAttribute, appropriatenessFactor). Here, ‘PA_B_x’ indicate the player type based on the classification of Bartle [2]. For example, ‘PA_B1, 0.9’ indicate that the GOB is very appropriate for the player type ‘Explorer’ according to Bartle.

The model for the learning context C_L provides a vector composed out of two parts listing all associated and prerequisite skills for a specific learning situation/LOB. In the example, ‘$A1_{xyz}$’ and ‘$B2_{xyz}$’ skills represent identifiers for learning topics of the 80Days´ demonstrator 1 DEG being extracted out of the curriculum for teaching geography in the 6^{th} to 8^{th} grade at school.

$$\left(\left\langle \begin{matrix} (\text{HJ\_2}, 0.1), \\ (\text{HJ\_4}, 0), \\ (\text{HJ\_4\_1}, 0.2), \\ (\text{HJ\_5}, 0.85), \\ \vdots \end{matrix} \right\rangle, \left\langle \begin{matrix} (\text{PA\_B1}, 0.15), \\ (\text{PA\_B2}, 0.4), \\ (\text{PA\_B3}, 0.2), \\ (\text{PA\_B4}, 0.9) \end{matrix} \right\rangle, \begin{pmatrix} \langle A1030, A1033 \rangle \\ \langle B2122, B2297 \rangle \end{pmatrix} \right)$$

Fig. 4. Quantifiable part of the model for Narrative, Game-based Learning Objects.

Apart from that quantifiable part described above, the model for NGLOB contains further descriptive elements such as short texts/abstracts summarizing the synopsis of narrative, gaming and learning functions of a specific NGLOB.

Figure 5 provides the preliminary technical implementation of the NGLOB model in form of an XML-based schema, which has been integrated into the ICML format. ICML (INSCAPE Markup Language) has been originated by the authors of this paper within the European research project INSCAPE[3] [5] and used and further cultivated in other projects such as U-CREATE[4] [10] or 80Days. In brief, the global aim of ICML is to provide a standardized comprehensive description language for Story-based DEG and *any* kind of Interactive Storytelling application. The ICML format [8] provides three top-level nodes: ICML_content, ICML_strategies and ICML_story. The model of NGLOBS is being integrated and serves as detailed annotation and specification of ICML_story parts (complex scenes, scenes ~ micro missions and situations in 80Days or game levels/learning chapters and game situations/learning units in games/learning applications).

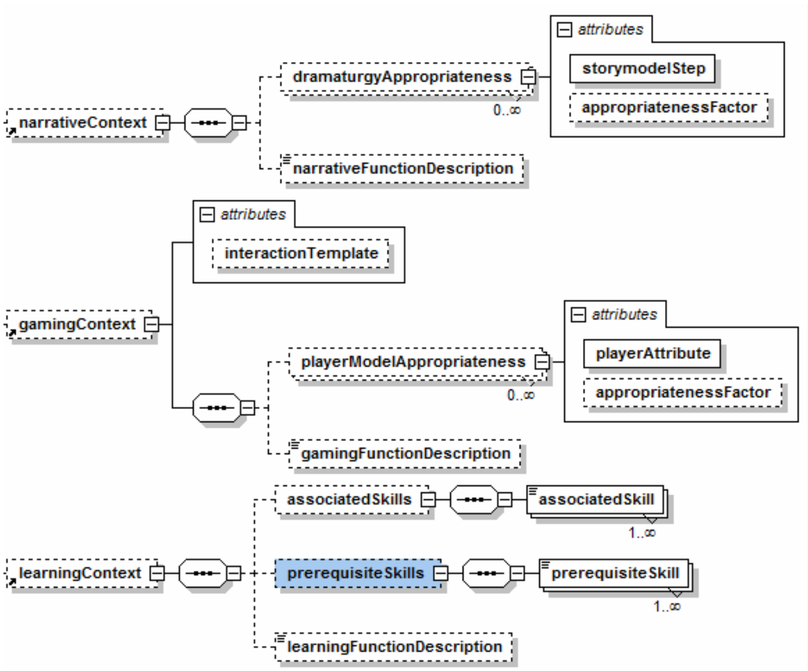


Fig. 5. XML-based Schema for NGLOB, being integrated within ICML_story

3 Narrative Game-based Learning Objects in 80Days

This section describes the creation and use of NGLOB in 80Days. Hereby, the StoryTec platform providing a complete framework consisting of an authoring tool and a run-time environment serves as technical basis. In [8, 9] Göbel et al. provides a comprehensive description of StoryTec. In 80Days, the authoring tool and the Story Engine are used and integrated into the 80Days framework.

[3] INSCAPE: Interactive Storytelling for Creative People (FP6, IP, IST-2004-004150), http://www.inscapers.com

[4] U-CREATE: Creative Authoring Tools for Edutainment Applications (EU CRAFT, COOP-CT-2005-017683)

In order to use NGLOB in Story-based DEG such as 80Days, StoryTec has been enhanced both in the authoring tool and run-time components: In the authoring tool, the Property Editor provides fields to enter context information of NGLOB. The Condition Editor is used to enter application logic and define conditions for transitions among story units (→ sequencing of story units in Story-based DEG, see also section 1).

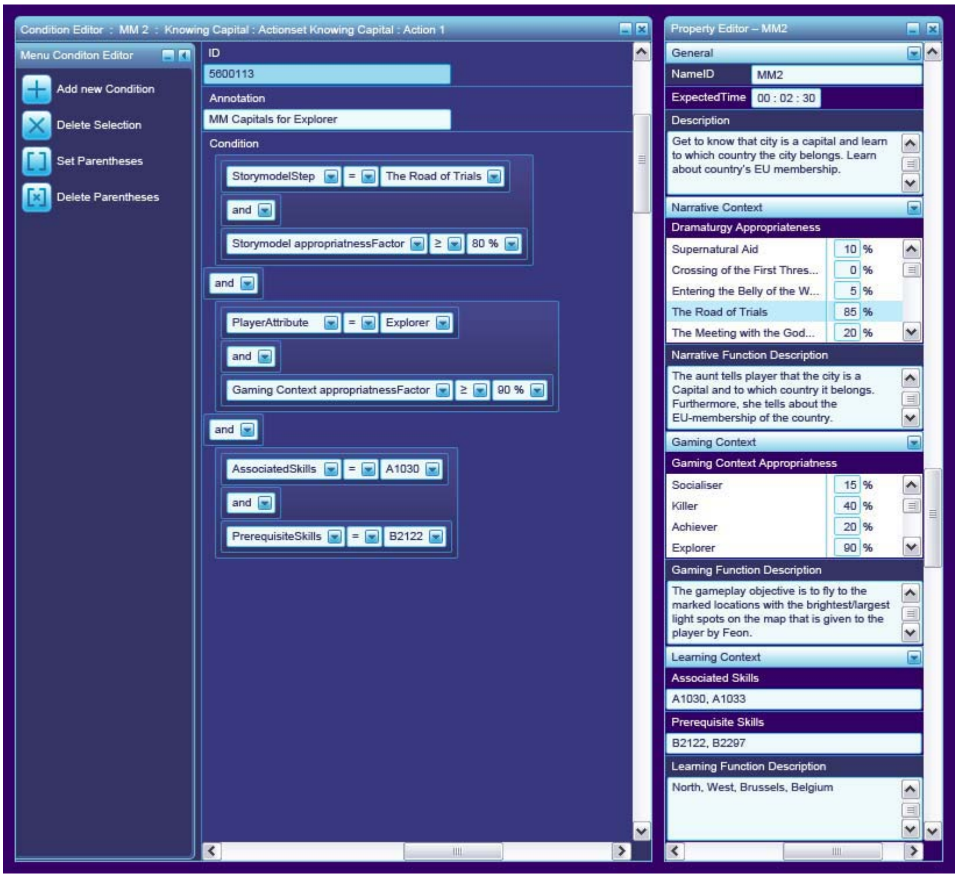


Fig. 6. Condition Editor (left) and Property Editor (right).

The result of the authoring process are ICML-encoded stories. These ICML files are loaded into the run-time environment of the 80Days framework in form of an executable story graph in the Story Engine. Then, the story starts and unfolds according to the user/player/learners´ interactions, behavior, rules and application logic provided by the author, and context information. The context information of NGLOB is hold within the content layer (repository, see lower part in figure 7) and the modeling part (see upper part in figure 7).

Concerning the control of Story-based DEG and sequencing NGLOB, the modeling part of the framework provides a player and learner model [4] plus a configuration component: Here, the author can provide priorities for narration, gaming and learning. Hence, possible conflicts in the sense of ‘it is possible to continue the story with NGLOB A, NGLOB B or NGLOB C’ (which all fulfill the conditions for transitions to following story units) practically do not occur.

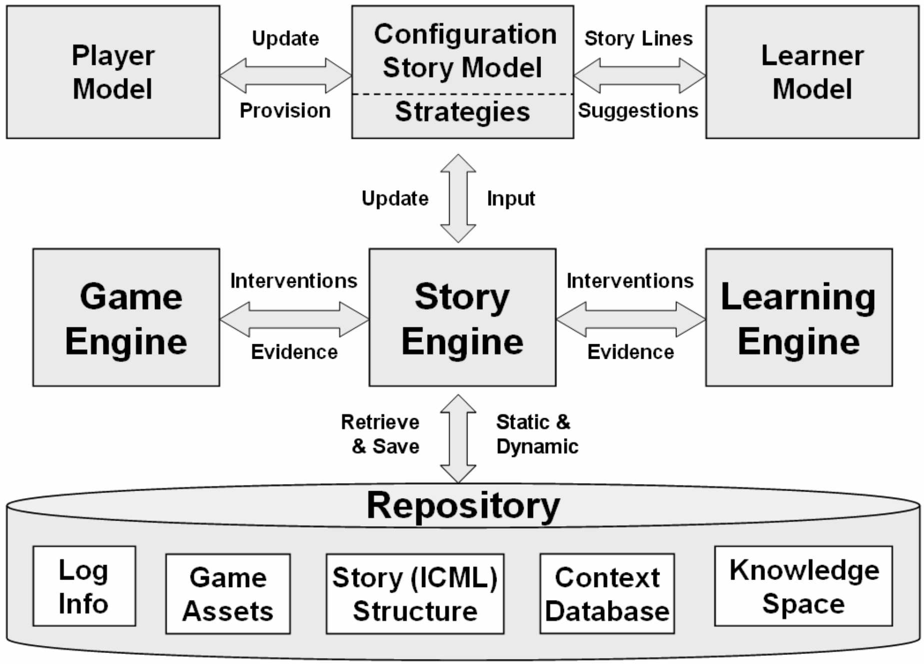


Fig. 7. High-Level Architecture of the 80Days Framework.

6 Conclusion

Based on preliminary, long-term research in the field of Story-based edutainment applications and DEG, this paper introduces the conceptualization of Narrative, Game-based Learning Objects (NGLOB) as basis for Story-based DEG. The main result represents a first version of a formalized model for NGLOB composed by narrative, gaming and learning context. The model has been technically implemented in the form of an XML schema being used as extension for the ICML format and has been rudimentary integrated into the StoryTec authoring environment and the 80Days run-time system.

Apart from typical testing evaluation studies, further research and development activities will be investigated a) to continue and enhance the model of NGLOB and b) to integrate methods and concepts of player and learner models.

Acknowledgements

The research and development introduced in this work is funded by the European Commission under the seventh framework programme in the ICT research priority, contract number 215918 (80Days, www.eightydays.eu).

References

1. Albert, D., & Lukas, J. (Eds.): *Knowledge spaces: theories, empirical research, and applications*. Mahwah, NJ: Lawrence Erlbaum Associates (1999).
2. Bartle, R. *Hearts, Clubs, Diamonds, Spades: Players Who Suit MUDs*. In The Journal of Virtual Environments, 1(1), 1996.
3. Campbell, J. *The Hero with a Thousand Faces*. 1st edition, Bollingen Foundation, 1949. 2nd edition, Princeton University Press. 3rd edtion, New World Library (2008).
4. Conloan, O., O´Keeffe, I., Hampson, C. and Heller, J. *Using Knowledge Space Theory to support Learner Modeling and Personalization*. In: T. Reeves & S. Yamashita (Eds.), Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2006, pp. 1912-1919. Chesapeake, VA: AACE (2006).
5. Göbel, S. and Becker, F. *INSCAPE – Interactive Storytelling for Creative People*. In Proceedings of EVA 2005 Berlin, pp. 136-141 (2005).
6. Göbel, S., Malkewitz, R., Becker, F.: *Story Pacing in Interactive Storytelling*. In Pan, Z., Aylett, R., Diener, H., Jin, X., Göbel, S., Li, L., eds. : Technologies for E-Learning and Digital Entertainment 3942. Springer, Berlin / Heidelberg (2006) 419-428
7. Göbel, S., Mehm, F., Radke, S.: *80Days: Adaptive Digital Storytelling for Digital Educational Games*. In: Proceedings of the 2nd International Workshop on Story-Telling and Educational Games (STEG'09). Edited by Cao, Y, Hannemann, A., Manjón, B.F., Göbel, S., Hockemeyer, C. and Stefanakis, E. Published on CEUR-WS:03-Sep-2009, http://CEUR-WS.org/Vol-498/ (2009).
8. Göbel, S., Salvatore, L., Konrad, R. *StoryTec: A Digital Storytelling Platform for the Authoring and Experiencing of Interactive and Non-linear Stories*. In Nesi, P., Ng, K. and Delgado, J.: Fourth International Conference on Automated Solutions for Cross Media Content and Multi-Channel Distribution, pp. 103-110. IEEE computer society (2008).
9. Göbel, S., Salvatore, L., Konrad, R., Mehm, F.: *StoryTec: A Digital Storytelling Platform for the Authoring and Experiencing of Interactive and Non-linear Stories*. In Spierling, U., Cavazza, M., Peinado, F., Aylett, R., Swartjes, I., Kudenko, D., Young, R., Tychsen, A., Pizzi, D., El-Nasr, M., eds. : Interactive Storytelling 5334. Springer, Berlin / Heidelberg (2008) 325-328
10. Göbel, S., Salvatore, L., Konrad, R., Sauer, S., Osswald, K.: *U-CREATE: Authoring tool for the creation of interactive storytelling based edutainment applications*. In Cappellini, V., Hemsley, J., eds. : EVA 2007 Florence Proceedings. Pitagora Editrice Bologna, Bologna (2007)
11. Hoermann, S., Hildebrandt, T., Rensing, C., Steinmetz, R.: *Resource Center - A Digital Learning Object Repository with an Integrated Authoring Tool Set*. In: Kommers, P., Richards, G., eds. : Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2005. AACE, Chesapeake, VA (2005) 3453-3460

12. Hoffmann, A., Göbel, S., Schneider, O. and Iurgel, I. *Storytelling-based Edutainment Applications*. In: Tan, Ledo (ed.): E-Learning and Virtual Science Centers. Hershey, Information Science Publishing (2005).

13. Kickmeier-Rust, M., Göbel, S., Albert, D.: *80Days: Melding Adaptive Educational Technology and Adaptive and Interactive Storytelling in Digital Educational Games*. In Klamma, R., Sharda, N., Fernández-Manjón, B., Kosch, H., Spaniol, M., eds. : Proceedings of the First International Workshop on Story-Telling and Educational Games 386. CEUR Workshop Proceedings (2008)

14. Kriegel, M., Aylett, R. *Emergent Narrative as a Novel Framework for Massively Collaborative Authoring*. In Gratch, J., Prendinger, H., Ishizuka, M., Marsella, S., Pelachaud, C., Kopp, S., Kipp, M., André, E., Rehm, M., Aylett, R., eds. : Intelligent Virtual Agents 5208. Springer, Berlin / Heidelberg (2008) 73-80

15. Louchart, S., Aylett, R. *Solving the narrative paradox in VEs – lessons from RPGs*. In: Intelligent Virtual Agents, 4th International Workshop IVA 2003. Eds. T. Rist, R. Aylett, D. Ballin. LNAI 2792, pp. 244-248. Springer (2003).

16. Louchart, S & Aylett, R.S. *Narrative Theory and Emergent Interactive Narrative*. International Journal of Continuing Engineering Education and Life-long Learning. Special issue on narrative in education. Vol 14 no 6, pp. 506-518 (2004).

17. Vogler, C. *The Writer´s Journey. Mythic Structure for Storytellers and Screenwriters*. Michael Wiese Productions, California (1998).

Undercover: Non-Invasive, Adaptive Interventions in Educational Games

Christina M. Steiner, Michael D. Kickmeier-Rust, Elke Mattheiss, and Dietrich Albert

University of Graz, Department of Psychology, Cognitive Science Section
Universitätsplatz 2, 8010 Graz, Austria
{chr.steiner, michael.kickmeier, elke.mattheiss, dietrich.albert}@uni-graz.at

Abstract. Computer games are extremely successful and popular – and the potential of using this medium for educational purposes is increasingly recognised and researched. To keep the engaging character of a computer game, an educational game needs to appropriately contextualise learning activities in the game and its narrative in order to retain flow and engagement of the gaming experience. The realisation of good educational games furthermore requires that such new learning technologies are appropriate for all students and ensure learning opportunities with individually appropriate levels of challenge, which calls for adaptation mechanisms to learners' abilities and motivation. In the present paper we present adaptation mechanisms that are strongly embedded into an educational game. Personalisation of learning and gaming experiences is realised in a two-fold manner, targeting a learner's competence as well as motivational state. The described non-invasive, adaptive interventions are researched, implemented, and evaluated in the context of the European research project 80Days.

1 Introduction

Information technology is an integral part of today's life and kids are nowadays familiar with the whole range of toys and tools of the digital age – computers, video games, internet, cell phones etc. Consequently, these children are – as characterised by Marc Prensky – no longer digital immigrants but rather digital natives [1]. This necessarily has also consequences for education; students of today are disengaged with traditional instruction – they are no longer the kind of students our educational system was designed for [2]. As a result, the application of computer technology for learning is growing. Aside from the development of general e-learning environments and computer-based trainings and courses, since the 1990s research has been increasingly dedicated to the use of computer games for learning and since then a widespread public interest has grown in using games as learning tools. On the one hand, this is inspired by the fact that playing is the most natural form of learning. On the other hand, the great interest in digital game-based learning is due to the popularity of computer games – digital games constitutes a multi billion industry per year and kids spend a considerable portion of their life- and free time in playing these games.

The design of good computer games for learning, however, constitutes a challenge. Not every educational game is necessarily good for learning – and not necessarily for all learners. The implementation of a successful and effective educational game needs to take care for an attractive and competitive game design while at the same time realising an appropriate learning design – both design aspects need to accompany and complement each other. The quality of an educational game can in general be maximised by having the game play done by game designers and the design of learning done by teachers [2].

One of the main reasons why games are assumed to be effective for learning is their engaging character. Games are able to induce what Csikszentmihalyi calls 'flow' [3] – a positively perceived experience and state of full immersion in an activity that typically goes along with a loss of sense of time. A successful educational game therefore needs to ensure to keep and promote this 'optimal experience' of flow. Anything that disrupts this experience and causes the player to 'leave' the current game situation should be avoided [2]. Because of that, conventional educational measures and activities as applied in classroom and 'traditional' e-learning environments, like intermediate explicit knowledge assessments, are not suitable in the context of educational games. Any time the player is forced to stop the game itself to do something else, flow is interrupted and thus, engagement, immersion, and motivation are compromised. As a consequence, the 'additional' activities that are due to the instructional character of an educational game need to be strongly embedded in the game such that the disruption of flow is minimised. This aspect of embedding instruction into the game experience and narrative is a crucial factor for realising good educational games and can be related to the educational conception of situated learning or cognition [4, 5]. Learning takes place in a situated context – learning events are embedded and contextualised in a meaningful situation of the game. The knowledge elements and skills targeted are relevant in the game context and are at the same time applied and practiced directly related to this environment [2, 6].

Another important aspect for effective game-based learning is the level of challenge an educational game imposes on the learner. An educational game needs to feature an appropriate difficulty level for the individual learner [2]. A game that is too easy or too difficult is not engaging for the player. Rather, an educational game should address the current capability of the learner by providing an appropriate level of challenge without exceeding his/her capacity to succeed and, in this way, retains motivation and engagement. Referring to Piaget's theory of cognitive development, Van Eck [2] aptly describes "games thrive as teaching tools when they create a continuous cycle of cognitive disequilibrium and resolution … while also allowing the player to be successful" (p. 20). As individual learners may largely differ with respect to their level of competence as well as their motivational constitution, effective game-based learning should make use of adaptation mechanisms that are able to personalise the learning experience to the individual learner. To this end, a game needs continuous input from the learner and to provide appropriate feedback and interventions. Due to the previously outlined claim for embedded and situated educational activities in the game, conventional adaptation technologies [7, 8] can sometimes hardly be directly adopted to the field of educational games.

In the present paper we describe adaptation mechanisms and principles for educational games, which implement personalised learning experiences that are embedded into the game situation. They ground on the non-invasive and continuous assessment of the learner's current competence and motivational state. By the use of adaptive interventions tailored to the information coming from this assessment a learner can be supported and guided in the game and motivation can be retained. In the following sections first the assessment procedures are briefly described. Subsequently, a detailed description of intervention types suitable for adaptation at the micro level of an educational game is provided. The 80Days project is presented, where the described research is implemented into an educational game teaching geography. We conclude with a short wrap up of the outlined work and future directions.

2 Non-Invasive Assessment

The basis of a non-invasive, continuous assessment of learning progress and motivational states is to monitor and interpret the learner's actions in the game. The respective assessment procedures ground on well-founded and elaborated psycho-pedagogical theoretical frameworks.

For knowledge and competence assessment the formal framework of Competence-based Knowledge Space Theory (CbKST) [9, 10] is utilised. Originating from conventional adaptive and personalised tutoring, this set-theoretic framework allows assumptions about the structure of skills in terms of underlying cognitive constructs of a knowledge domain and to link these latent skills with observable behaviour. Hereby, the entities (i.e. skills) of a knowledge domain are structured by the use of a well-defined relationship, the so-called prerequisite relation. The collection of subsets of skill corresponding to this prerequisite relation makes up the so called competence structure and characterises meaningful learning paths. Combining the competence structure with the problems or tasks of a learning situation the relation to observable behaviour can be established, such that from a learner's observable behaviour or actions inferences on his/her available and lacking skills can be made. While on principle, the CbKST framework builds the basis for an effective adaptive skill assessment that is carried out in form of explicit testing procedures [11], in a game context this assessment needs to be realised on a non-invasive micro-level, in a problem solving situation that is embedded into the game context and narrative [12]. A simple example for such a task in the context of an educational game may be to fly with a space ship to a certain city and to take a picture. The learning objective of this task might be (among others) to learn about the location of the city on the map. In this situation there are various manipulable objects, for example the space ship. The learner can perform certain actions to achieve the goal, in this example primarily changing the directions while flying or controlling speed and altitude. The aim of micro level assessment is in the first instance to assign a problem solution state from the problem space to each action (e.g., pressing an arrow key). This mapping is done by classifying actions according to a set of rules. An example for such rule might be 'the distance between space ship and target location is increasing'. The second aim is to assign a set of available and a set of lacking skills to each problem solution state; for example, flying in the right direction indicates that the learner knows the wind direction towards the city. Of course, a single observation is not very convincing. Thus, CbKST provides a probabilistic approach to assessment. We have a probability distribution over all possible competence states and with each action we update the probabilities of those states that include the relevant skills and we decrease those states that include the lacking skills. At the end of this procedure stands a more or less well-founded assumption about the skills the learners have, the skills they don't have, and their position in the problem solving process.

The non-invasive assessment of the learner's motivational state is grounded on an advanced model of motivation for educational games (see Fig. 1, [13]) that fuses several established theoretical approaches to motivation and learning [3, 14, 15, 16]. Similar to the non-invasive skill assessment sketched above, we can assign specific motivational assumptions to specific classes of actions. Certain behaviour patterns and characteristics, like e.g. mouse movements, time measurements, help demands etc., can be utilised as indicators for certain aspects of a learner's motivational state. If, for instance, the mouse movements of a learner are random, this can be interpreted as a lack of attention, and if hints are demanded frequently this can under certain conditions be a sign for a lack of confidence.

The continuously gathered and updated assumptions on the skills and motivational state throughout the game serve the provision of adaptive interventions tailored to the learner's current needs.

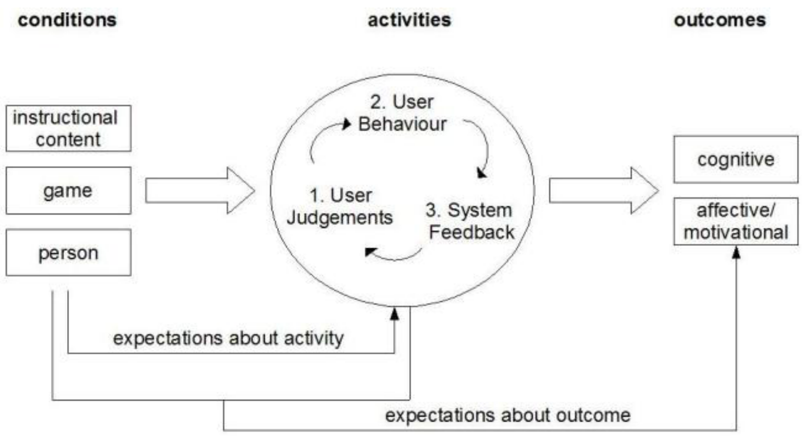


Fig. 1. Advanced model of motivation for educational games.

3 Adaptive Interventions

Depending on the information stemming from the non-invasive assessment, from a menu of adaptation types a system feedback in terms of an intervention in the game can be triggered (e.g. hints or suggestions through a non-player character, modification of display or interface) that is individually appropriate for the respective learner and situation. All types of adaptive interventions have in common that they aim at supporting a beneficial game-based learning experience. In general, the two broad categories of cognitive and motivational interventions can be distinguished, aligned with the two perspectives of non-invasive assessment.

3.1 Cognitive Interventions

Interventions of this type strive to enhance cognitive abilities and to support the learner adaptively according to his/her task behaviour and underlying available or lacking skills. Consequently, these interventions target the learning objectives defined in terms of skills and foster their successful acquisition in terms of prompting reflection or assisting the learner. The cognitive interventions and their selection rules are defined based on cognitive and psycho-pedagogical theories and considerations – they relate among others to the paradigm of self-regulated learning [17] and the importance of reflection on the task and oneself for effective learning, to the CbKST framework and the derivation of meaningful learning paths based on its evolving competence learning structures [10], as well as to theoretically founded principles for the design of informative tutoring feedback [18]. The following cognitive intervention types can been distinguished, whereby the line between the different types is partly somewhat blurred:

- **Meta-cognitive interventions** are supposed to provoke learners' reflection about their own abilities, thinking processes, solution behaviour, or confidence and may consist in metacognitive questions or tasks and certainty questions (e.g. 'Does this solution make sense?', 'How sure are you about this?').
- **Competence activation interventions** are applied if a learner becomes stuck in a certain task while foregoing assessment results led to the assumption that the learner possesses the necessary skills. By the use of an appropriate intervention (e.g. 'We have come across this issue already before.') the temporarily 'inactive' skills are assumed to be stimulated and reactivated.
- **Competence acquisition interventions** are selected when the system concludes that a learner lacks certain skills and thus, provide the required information – for example through a non-player character.
- **Problem solving support** is provided in the context of an ongoing problem solving process and provides hints and indications for possible next problem solution steps in order to decrease the distance between the present solution state and the target state.

- **Dissolving interventions** are a further form to present specific information to the learner. The purpose of this intervention type is to provide the solution of a problem/task if the learner was not able to show the required answer behaviour within a reasonable number of actions. Such interventions, ultimately, shall assure that the game can continue and thus the gaming and flow experience are kept going.

- **Progress feedback** is made up by interventions that provide the learner with information about the learning progress or the game – e.g. through a non-player character or different scoring mechanisms – and thus foster monitoring and reflection on one's own performance.

- **Cognitive assessment interventions** are a special form of intervention that is applied if the non-invasive assessment of skills led to unclear or ambiguous results after a certain number of actions. In order to gather additional information and improve the assessment this type of intervention is triggered. Typically this is realised by providing the learner with explicit questions or problems. As these interventions are strongly embedded into the game context and narrative, they differ significantly from conventional and possibly disrupting pop-up assessments known from 'traditional' e-learning. Assessment interventions may be realised by interactive dialogues with different answer options, which may not only refer to correct and incorrect responses but may also have a storytelling function and lead to different story strands depending on the learner's choice.

3.2 Motivational Interventions

Motivational interventions are supposed to enhance and retain the learner's motivation and engagement on a high level or to intervene when the system detects that the motivational state or certain aspects of it decrease. The differentiation of motivational intervention types is inspired and their selection rules are defined based on psycho-pedagogical theories on motivation and motivational design, such as the expanded model of motivation to learn [16], attribution theory [19] and the concept of self-efficacy [20], and Keller's ARCS model [14]. The following intervention types are distinguished:

- **Praising interventions** are used for congratulation in case of success.

- **Encouraging interventions** are applied especially in case of failure in order to promote further trials.

- **Attributional interventions** go further than the previously mentioned interventions – they aim at fostering self-worth enhancing attributional styles for success and failure and are applied in case of lacking confidence or dysfunctional attributional styles. This is realised by a motivational training based on attribution theory [19, 21] – in form of feedback that directs attribution of success to internal factors (i.e. effort and ability) and attribution of failure to variable components (i.e. lack of effort and bad luck).

– **Incitation interventions** in general announce pleasing outcomes like rewards in order to foster motivation to carry on in the game or to proceed in a problem solving situation.

– **Affective interventions** address emotional-affective aspects of the game experience and social interaction with other game characters and are supposed to foster a positive affect.

– **Attention-catchers** are interventions that are applied if the system detects decreasing or lacking attention through the interpretation of the learner's actions. Such interventions constitute unexpected changes or incidents and in this way increase variability and further appeal of the game.

– **Motivational assessment interventions** are similar to their cognitive counterparts. They are utilised in case of inconclusive or contradicting inferences on the learner's motivational state based on the non-invasive assessment. For gathering further indications on the learner's current motivation assessment interventions realise an explicit questioning, usually in form of an interactive dialogue with a non-player character and with the answer options relating to certain aspects and states of motivation.

4 Putting it into Practice: 80Days

The research, elaboration, and technical implementation of non-invasive assessment of learners' competence and motivation, as well as of adaptive interventions and adaptation principles in the context of educational games are addressed in the 80Days project (www.eightydays.eu). 80Days is a European research project aiming at advancing psycho-pedagogical and technological foundations for successful digital educational games through the development of a higher-level theoretical framework for adaptive educational technology. This shall allow an adaptation of a game's story and features to individual learners' abilities, engagement, and preferences.

Inspired by Jule Verne's novel 'Around the world in eighty days' an educational game is implemented that constitutes a modern version of a journey around the world – in a UFO with an alien travel companion (see Fig. 2). From an educational perspective, the game's main objective is to teach geography skills. From a storytelling perspective, the main task for the player is to explore the planet and collect information for an intergalactic travel guide. From the game play perspective, the main goal is to navigate the UFO to different destinations around the world and to accomplish a variety of adventurous missions.

The game incorporates adaptation to the individual learner through continuous, non-invasive assessment procedures and the realisation of interventions matching the learner's current level of knowledge and retaining motivation. All interventions of the game require manifestation in form of game assets. A main issue is the translation of the general, theory-based selection rules for interventions into associated triggers within the game context, such that appropriate interventions are provided in an appropriate extent and appropriate point in time. This is a challenge especially as the repeated and/or inadequate provision of interventions and misinterpretation of situations and actions can be counterproductive and do considerable harm to motivation, engagement, and flow. The realisation of repeated design and development cycles of successive demonstrator game releases and aligned evaluation cycles allow the continuous elaboration and refinement of the underlying adaptation mechanisms.

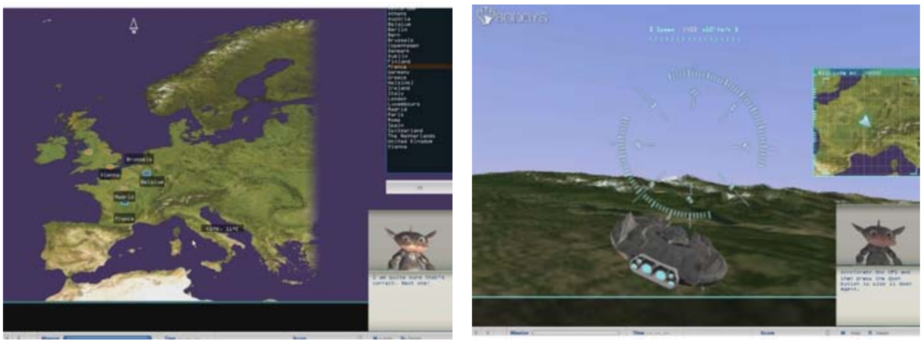


Fig. 2. Screenshots of the 80Days demonstrator game.

5 Conclusion and Outlook

Computer games are tremendously successful and popular and in recent decades an increasingly widespread public interest has evolved in using this very technology for educational purposes. Games' potential of being effective learning tools is appreciated because of their engaging character and their active and dynamic nature. The main challenge of realising successful educational games is to intertwine an attractive game design with a sound didactic design that realises learning embedded in the game's situations and context – instead of having playing interrupted by learning activities, which would compromise flow and engagement considerably. Another critical issue is an appropriate level of challenge that should be imposed to the learner in order to realise an engaging gaming and learning experience. To suit different learners an educational game therefore should continuously adapt to an individual learner's knowledge and motivation. To come up to this necessity common adaptation technologies as used in conventional educational systems are generally not sufficient and/or suitable; rather, assessment procedures and adaptive interventions are needed that are strongly integrated into the game – in order to enable learning in a situated context, as mentioned above.

In the present paper we have presented adaptive interventions that are suitable for personalisation and learning embedded in an educational game. The purpose of these adaptive interventions is two-fold, on the one hand addressing the learning aspect and on the other hand addressing motivation. For each of these two categories different intervention types and according selection rules can be defined based on cognitive and psycho-pedagogical theories and considerations. The adaptive interventions ground on assumptions on the currently available skills and motivational state of a learner based on a continuous, non-invasive assessment and interpretation of the learner's actions in the game. A precondition for realising successful interventions is therefore a successful assessment process yielding valid assumptions on the learner's characteristics on which the interventions are relying. This calls for valid skill structures and skill assignments to tasks, for appropriate indicators of motivational aspects, as well as for the proper interpretation of learner actions. The probabilistic assessment of skills is not perfect, thus it is sometimes reasonable to strengthen the conclusions drawn by more explicit information like cognitive assessment interventions. In case of the assessment of motivational aspects especially the selection and operationalisation of suitable indicators is demanding and complicated. The frequent use of help functions, for example, might be interpreted as a lack of confidence, but may also result from help abuse and so-called 'gaming the system'. Therefore, it is necessary to carefully define the indicators and rules for drawing assumptions on the motivational state. Often the triangulation of different indicator variables will be advisable.

The outlined mechanisms of adaptation are researched and implemented in the course of the 80Days project, which aims in advancing intelligent and competitive educational games on a European level. In the context of this project and its predecessor ELEKTRA (www.elektra-project.org) empirical investigations have been initiated in order to investigate the empirical effectiveness of adaptive features in assessment and interventions. Early analyses revealed that adaptation results in better learning performance and superior game experience than it was the case in non-adaptive control groups [22]. Future research needs to address in detail the different intervention types and whether they yield the expected benefits. In addition, the considerations on motivational assessment and adaptation and the underlying advanced model of motivation in educational games need to be further investigated. The in-depth empirical evaluation of the 80Days demonstrator games will serve further refinements of the theoretical framework for the adaptation mechanisms and improvement of the game and didactic design.

The focus of our work so far was on elaborating suitable methods for providing learners with tailored psycho-pedagogical guidance and support that is strongly embedded in the game – thus realising learning in a situated gaming context, with an appropriate level of challenge, and minimising interruption of flow. A future direction of your research will be to enter also the level of collaborative learning and to apply the principles of non-invasive, adaptive interventions on the group level. To this end, the principles for personalisation and adaptation developed and implemented for individual learners need to be further advanced to take care for the specificities of collaborative learning. This needs to take into account and synthesise state of the art on computer supported collaborative learning (e.g. [23]), (massively) multiplayer games (e.g. [24]), group adaptation (e.g. [25]), as well as psychological theories of groups and group learning (e.g. [26, 27]).

Acknowledgements

The research and development introduced in this work is funded by the European Commission under the sixth framework programme in the IST research priority, contract number 027986 (ELEKTRA, www.elektra-project.org) as well as under the seventh framework programme in the ICT research priority, contract number 215918 (80Days, www.eightydays.eu).

References

1. Prensky, M.: Digital Natives, Digital Immigrants. On the Horizon 9 (2001) 1–2
2. Van Eck, R.: Digital Game-Based Learning. It's Not Just the Digital Natives Who Are Restless. Educause Review (2006) 16–30
3. Csikszentmihalyi, M.: Flow: The Psychology of Optimal Experience. Harper Perennial, New York (1990)
4. Lave, J., Wenger, E.: Situated Learning: Legitimate Peripheral Participation. Cambridge University Press, New York (1991)
5. Cognition and Technology Group Vanderbilt: Anchored Instruction and Situated Cognition Revisited. Educational Technology 33 (1993) 52–70
6. Van Eck, R., Dempsey, J.: The Effect of Competition and Contextualised Advisement on the Transfer of Mathematics Skills in a Computer-Based Instructional Simulation Game. Educational Technology Research and Development 50 (2002) 23–41
7. Brusilovsky, P.: Adaptive and Intelligent Technologies for Web-based Education. Künstliche Intelligenz 4 (1999) 19–25
8. DeBra, P.M.E.: Adaptive Hypermedia. In Adelsberger, H., Kinshuk, Pawlowski, J.M., Sampson, D. (eds.): Handbook on Information Technologies for Education and Training. Springer, Berlin (2008) 28–35
9. Doignon, J.-P., Falmagne, J.-C.: Knowledge Spaces. Springer, Berlin (1999)
10. Albert, D., Lukas, J.: Knowledge Spaces: Theories, Empirical Research, and Applications. Lawrence Erlbaum Associates, Mahwah, NJ (1999)
11. Dowling, C.E., Hockemeyer, C.: Automata for the Assessment of Knowledge. IEEE Transactions on Knowledge and Data Engineering 13 (2001) 451–461
12. Kickmeier-Rust, M.D., Hockemeyer, C., Albert, D., Augustin, T.: Micro Adaptive, Non-Invasive Assessment in Educational Games. In Eisenberg, M., Kinshuk, Chang, M., McGreal, R. (eds.): Proceedings of the Second IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning (2008) 135–138
13. Mattheiss, E., Kickmeier-Rust, M., Steiner, C., Albert, D.: Motivation in Game-Based Learning: It's More than ‚Flow'. Paper accepted for presentation at the Game-Based Learning Workshop at DeLFI (2009)

14. Keller, J.M.: Development and Use of the ARCS Model of Motivational Design. Journal of Instructional Development 10 (1987) 2–10
15. Garris, R., Ahlers, R., Driskell, J.E.: Games, Motivation, and Learning: A Research and Practice Model. Simulation & Gaming 33 (2002) 441–467
16. Heckhausen, J., Heckhausen, H.: Motivation und Handeln [Motivation and Action]. Springer, Heidelberg (2006)
17. Zimmerman, B.J.: Becoming a Self-Regulated Learner: An Overview. Theory into Practice 41 (2002) 64–70
18. Narciss, S., Huth, K.: How to Design Informative Tutoring Feedback for Multi-Media Learning. In Niegemann, H.M., Leutner, D., Brunken, R. (eds.): Instructional Design for Multimedia Learning. Waxman, Munster, NY (2004) 181–195
19. Weiner, B.: Achievement Motivation and Attribution Theory. General Learning Press, Morristown, NJ (1974)
20. Bandura, A.: Self-Efficacy: Toward a Unifying Theory of Behavioral Change. Psychological Review 84 (1977) 191–215
21. Dresel, M., Ziegler, A.: Langfristige Förderung von Fähigkeitsselbstkonzept und impliziter Fähigkeitstheorie durch computerbasiertes attributionales Feedback [Long Term Advancement of Competence Self-Concept and Implicit Competence Theory by Computer-Based Attributional Feedback]. Zeitschrift für Pädagogische Psychologie 20 (2006) 49–63
22. Kickmeier-Rust, M.D., Marte, B., Linek, S.B., Lalonde, T., Albert, D.: The Effects of Individualized Feedback in Digital Educational Games. In Conolly, T., Stansfield, M. (eds.): Proceedings of the 2nd European Conference on Games Based Learning. Academic Publishing Limited, Reading, UK (2008) 227–236
23. Stahl, G., Koschmann, T., Suthers, D.: Computer-Supported Collaborative Learning: An Historical perspective. In Sawyer, R.K. (ed.): Cambridge Handbook of the Learning Sciences. Cambridge University Press, Cambridge, UK (2006) 409–426
24. Yee, N.: The Demographics, Motivations, and Derived Experiences of Users of Massively Multi-user Online Graphical Environments. Presence 15 (2006) 309–329
25. Vasilyeva E., Pechenizkiy M., DeBra P.: Adaptation of Feedback in e-learning System at Individual and Group Level. In: Proceedings of the International Workshop on Personalization in E-Learning Environments at Individual and Group Level (2007) 49–56
26. Sader, M.: Die Psychologie der Gruppe [The Psychology of Groups]. Juventa Verlag, München (1998)
27. Shlomo, S. (ed.): Handbook of Cooperative Learning Methods. Praeger Publishers, Westport (1999)

Towards Intelligent Tutoring Systems Based on Computer Role-Playing Games

Dennis Maciuszek and Alke Martens

University of Rostock, Institute of Computer Science, Albert-Einstein-Str. 21
D-18059 Rostock, Germany
{dennis.maciuszek, alke.martens}@uni-rostock.de

Abstract. Commercial educational computer games have not yet succeeded in integrating gameplay and learning in a satisfying way. This task is difficult, as well, for more advanced academic prototypes of adaptive educational games aka game-based intelligent tutoring systems (ITS). A variety of game-based ITS architectures have been proposed over the years, each introducing their own, more or less artificially constructed solutions of the game-learning integration task. While interesting and promising in themselves, no approach has emerged as a de-facto standard – as for non-game ITS Clancey's generic architecture – or been reused in a larger number of projects. To work towards a more generic and reusable solution, we therefore go back to the Clancey architecture and find a more natural way of transforming it into a game. We outline an ITS architecture based on computer role-playing games, indicate how it can be realised using software components, and sketch an example application.

1 Introduction

Learning and playing go together naturally (e.g. [15,20,37]). Obviously, children learn by playing. Yet, as of today, commercial educational computer games for children, youths, and adults have not succeeded in integrating gameplay and learning in a satisfying way.

In an experiment using eyetracking technology and logfile analysis, Bormann et al. [2] evaluated the educational adventure game *Physikus* (BrainGame, 1999), which consists of an educational part and a game part. These are linked via puzzles in the game that require the player to apply knowledge from the educational part. The study found that in the educational part learners focused mostly on immediately game-related information, and overall, learners spent much more time playing the game than reading the educational material.

Bormann et al. suspect that a physics game like *Genius – Tech Tycoon* (*Genius – Unternehmen Physik*, Radon Labs, 2004), which does not explicitly separate the game from a glossary, would perform better. However, as Jantke [21] points out, content in the *Genius* series is just as separated, here into a strategical simulation game and eLearning exercises. Both are not related by means of content, only by means of gameplay. In the physics exercises, the learner must earn money to spend in the strategy game. *Genius Biology* (*Genius – Task Force Biologie*, Radon Labs, 2005), on the other hand, according to Jantke's experiences, can be completed without ever looking at the exercises.

It should be noted that the mentioned games sold well and won a number of awards, so despite their many weaknesses they are among the best available.

Another promising educational game is *Global Conflicts: Palestine* (Serious Games Interactive, 2007). The player acts as a journalist investigating the Isreali-Palestinian conflict. He or she may choose either side or work for a neutral European paper. Meeting residents from both cultures, he or she eventually constructs articles based on the interviews. Harr, Buch, and Hanghøj [18] evaluated the game, and found more discrepancies between playing and learning. Writing virtual newspaper articles does not allow for enough freedom to express own critical thinking, which is what the player is supposed to learn. Still, players enjoyed being a journalist so much that job aspects dominated the political content.

The game-learning integration task is difficult, as well, for more advanced academic prototypes of adaptive educational games aka game-based intelligent tutoring systems (ITS). The idea of integrating intelligent tutoring with game architectures can be traced back to the early days of ITS research. In 1982, Burton and Brown [5] added an intelligent tutor to a board-game-like digital educational maths game. The gameplay and the maths domain together basically work as the knowledge domain of the ITS. Over the years, and especially now in the 2000s, when game-based learning received more attention, a variety of more advanced game-based ITS architectures have been proposed. Each of these architectures introduces its own, more or less artificially constructed solution of the game-learning integration task.

A number of approaches [4, 9, 33, 36, 39] still keep the game (or interactive storytelling system) and the intelligent tutor separate, but establish well-defined interfaces for exchanging data between both parts. Siemer and Angelides [38] embed an ITS in an existing game, whereas Gómez-Martin, Gómez-Martin, and González-Calero [16] use a game or story thread as the connecting element of their ITS lessons.

The *80Days* project calculates a product of their ITS knowledge structure (a competence space from competence-based knowledge space theory) and their three-act game story (the different plot paths possible in an interactive storytelling environment). While this is a mathematically sound approach instead of mere software-engineering heuristics, it potentially increases the size of the expert knowledge model by a lot.

In another solution, Carro et al. generate a game-based learning environment adaptively from a repository of building blocks. Further approaches take a distinct part of the ITS, and replace it by a game part. In [31], the game is the user interface of the ITS. The system in [24] replaces an existing learning environment by a game environment such that the tutor appears in the game.

All of these architectures are interesting and promising in themselves. A broad variety is good, but it is also puzzling that since the early 1980s no direction has emerged as a quasi-standard or at least been reused and evaluated in a larger set of projects. To work towards a more generic and reusable software architecture, the following section goes back to the quasi-standard according to which most non-game ITS are being designed today (consciously or not). We will propose a natural way of transforming this architecture into a game engine without a need for separate systems, embedding systems in one another, or calculating a cross product of game and ITS. Section 2 outlines an ITS architecture based on a certain game genre, and indicates how it can be realised using software components. Section 3 sketches an example application. Section 4 discusses what this approach might mean for the learner modelling experience of the player. Section 5 concludes, and points out further steps.

2 Proposed Architecture

Intelligent Tutoring Systems can look back on a comparably long tradition. First described in the late 1970s, they have been influenced by research in cognitive sciences, in artificial intelligence, and in computer science (in general). One of the first who described the underlying architecture of ITS has been Clancey [8]. This basic architecture has grown over the years and has been extended stepwise. The first Intelligent Computer Aided Instruction Systems consisted mainly of expert knowledge and e.g. dialogue techniques for interaction with the learner (see e.g. [35] and [27]). Later, ITS have been developed, where the expert knowledge has been enriched by pedagogical knowledge (e.g. [27]). Furthermore, the learner model has been integrated into the architecture. All these developments led to the current state of the ITS architecture.

The parts of the ITS architecture are: the user interface model, the learner model, the expert knowledge model, the pedagogical knowledge model (Fig. 1). The names of the parts vary, as well as the main functionality and the role of each of the parts – however, the parts can still be located in almost all existing ITS. A more detailed analysis can be found for example in [30].

Our research within game-based learning focuses on the objective of better integrating playing with learning. One approach to this problem is to look at *genres*. Game genres are defined by a common gameplay (i.e. the game mechanisms and the challenges for the player). Game genres include e.g. puzzle games, adventure games, or strategy games. When we aim at transforming a certain eLearning paradigm into a game, we look for a genre that matches the chosen eLearning model naturally. Ideally, there would be a one-to-one mapping of concepts. Our claim is that this facilitates the game-learning integration. For instance, we have previously argued that (and how) virtual laboratories can be made into simulation games [28].

Looking at ITS and adaptivity, one can notice a strong resemblance to the genre of computer role-playing games (RPG). Koubek [26] points out the connection between RPGs and adaptivity in serious games, but does not relate it to an architecture or ITS. Educational RPGs have been developed by researchers, e.g. [6], but do not exploit their adaptivity for tutoring purposes.

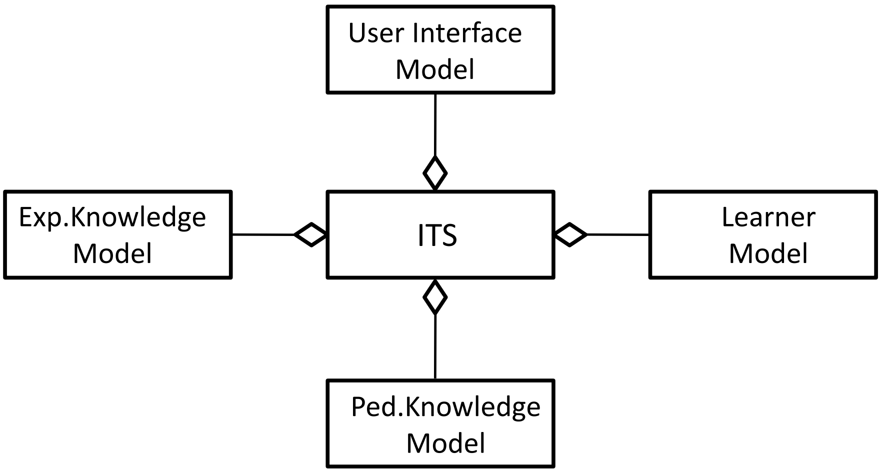


Fig. 1. Generic ITS architecture

RPGs are based on a non-computer type of game, pen-and-paper role-playing games. In this paper, we strictly focus the discussion on these *Dungeons & Dragons* types of games, or rather their computer-based successors. Another kind of "role-playing game", like for example the beer-distribution simulation game [23], which is used in education in business schools or in law, is not meant here. We are not talking about pedagogical role playing for training the resolution of social conflicts, either.

According to Hallford and Hallford – who wrote 'the' book on computer RPG design [17] –, the defining ingredient of a game in this particular genre is that the player develops the role of his or her player character (PC). Also called an avatar, the PC is the on-screen alter ego of the player. He or she may generate the PC by making lots of detailed choices about the avatar's appearance, abilities, backstory etc. A simpler approach is to select from pregenerated classes, e.g. warrior or wizard. In a third version, which Hallford and Hallford call the foundling approach, the player is simply assigned an avatar. In many games, the player can actually choose which method to follow. The RPG enthusiast will have fun specifying every detail of the avatar. A less experienced player may wish to answer a couple of questions embedded in a story, and let the system choose a class based on the answers. A player who would like to jump into the game directly can just start with a foundling.

There would still be user-determined character development during the game, because as the story advances and the player leads the character from quest to quest, the virtual person gains experience, and improves its abilities. It learns! What the character can learn is determined by the rules system of the computer RPG. For instance, in the science fiction RPG *Star Wars: Knights of the*

Old Republic (BioWare, 2003), a PC's abilities are defined by attributes (e.g. Strength, Dexterity), skills (e.g. Persuade, Repair), feats (special abilities like Empathy or Sneak Attack), and the overall experience level. These are recorded on a character sheet, mostly as numerical values. This is a learner model! The different forms of character creation are like different forms of ITS assessment.

Furthermore, there are prerequisites for the PC's learning. For instance, to learn Improved Empathy in *Knights of the Old Republic*, one needs to have Empathy and have reached an overall experience level of 4. A number of computer RPGs use somewhat more complex rules systems, e.g. *The Dark Eye: Drakensang* (Radon Labs, 2008). Many pen-and-paper RPGs use much more complex prerequisite structures, such as the magic system in *GURPS* (Steve Jackson Games, 1986–). The rules system is an expert knowledge model!

In a pen-and-paper RPG, a number of players play together (this is actually reflected in many computer RPGs by having a group or "party" of PCs, and by the sub-genre of massively multiplayer online RPGs = MMORPGs). One of these persons is the gamemaster. His or her role is to present the interactive story (a quest), to portray the non-player characters (NPCs), and through conversation and gaming aids like maps, miniatures, or models to simulate the commonly imagined environment. Notice that the gamemaster adapts the content of the story to the player characters.

In a computer RPG, a drama manager (this is a term from the field of interactive storytelling [10]) controls these aspects. Quests correspond to problems, scenarios, or cases in constructivist eLearning scenarios. NPCs may be enemies, but also employers, mentors, or informers. These latter roles correspond to the actual intelligent tutor, i.e. such NPCs can pose domain problems to solve, and give adaptive feedback or clues.

The environment in an RPG is the workplace of the learner in an ITS. For instance, this might be a laboratory in a chemistry program. Many fantasy RPGs feature alchemy labs, tools, recipes, as well as character skills to use these. In a language training course, the 'workplace' might be a foreign country that is to be explored. This virtual workplace needs to have a graphical representation – in an eLearning software as well as in an RPG. Nowadays, both of these are often 3D worlds. The multiuser virtual environment *Second Life* (Linden Lab, 2003–; see [28] for a discussion) actually looks like an MMORPG, but is now heavily used for online learning sessions. In conclusion, an RPG engine would contain all four parts of the generic ITS.

Fig. 2 shows a high-level version of our proposed RPG arcitecture. It takes Clancey's four ITS parts, and replaces them with their game equivalents. Notice that the pedagogical component aka the drama manager is divided into three sub-components (modules). Simply put, if the architecture was realised as a game engine, the teacher/game designer would just have to insert the training content as they would insert it into the ITS architecture.

The main parts of the traditional ITS (Fig. 1), in combination with the insights provided by research in software development, in particular object-oriented design, led to the Plug 'n Train framework (similar to [19]). The underlying

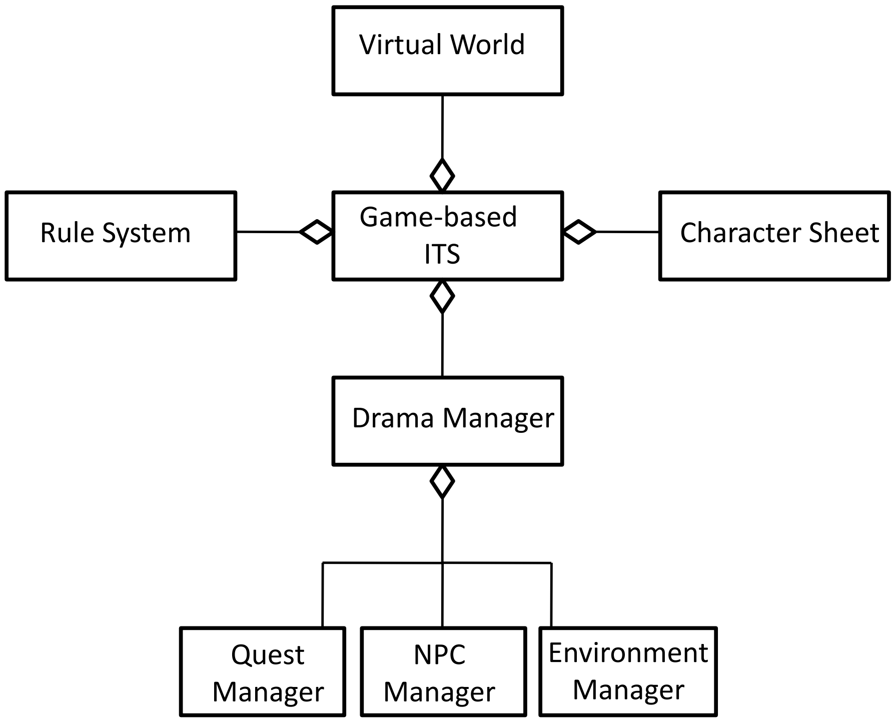


Fig. 2. Game-based ITS architecture

metaphor of this framework is to have a reusable set of main components, which are extended by modules and implemented by communication via interfaces. New modules can be plugged into existing components. Intermodular communication is provided by a network of observers. A detailed description of the plugin framework can be found in [32]. The framework is depicted in Fig. 3.

The Plug 'n Train framework is implemented using Java and XML, and we are currently developing eLearning and game-based learning components. Since it is especially suited for creating ITS, we can now also use it to start designing an RPG engine consisting of the four components, divided into seven modules as depicted in Fig. 2. The following section sketches an example instantiation.

3 Example Design

In this section, we start specifying modules of an hypothetical educational RPG. Note that we do not specify game content and educational content sepearately, but rather at the same time. The point is to find natural, one-to-one game representations of the items in the knowledge domain.

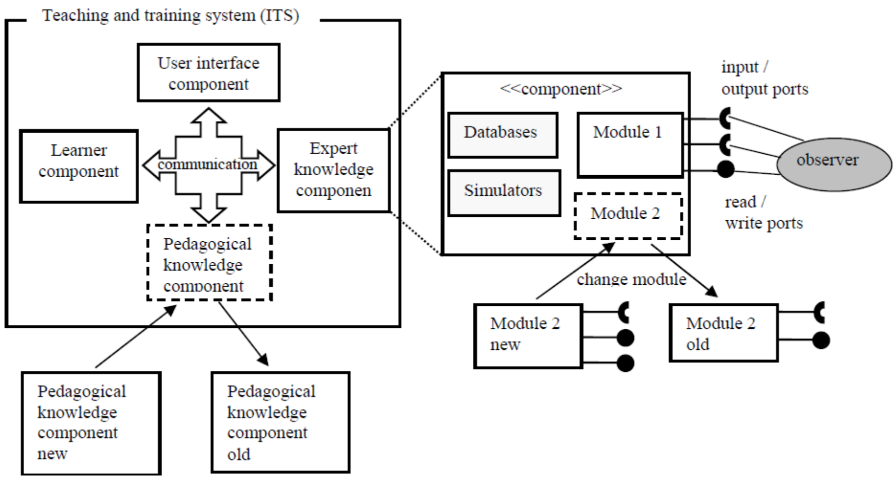


Fig. 3. Component-based ITS framework [32]

For this example, we chose a sub-domain of artificial intelligence (AI) for software agents – something we are currently teaching to third-year computer science students. The sub-domain is called steering behaviours. These behaviours enable an agent to do things like pursuing another agent or avoiding obstacles. Possible application areas are in game AI, simulation of animals (flocks of birds or schools of fish), or robotics.

Steering algorithms were invented by Reynolds [34], and are documented in detail e.g. in [3]. Here, we use the latter source for deriving knowledge items and a relation of prerequisites. For the sake of simplifying the example, we use only a subset of the available algorithms here. All steering algorithms are vector-based, as they compute forces and impulses in 2D or 3D space. Therefore, any learner studying this topic will also need a working knowledge of linear algebra.

In order to define knowledge items in the expert knowledge and learner model as well as semantic relationships between those – here, exclusively prerequisite relationships –, we apply a mathematical modelling mechanism for ITS design: basic knowledge space theory (KST) [11, 12]. The *80Days* project uses an advanced version of this theory [25]. This may or may not become the ultimate choice for the modelling used in our systems, yet for a demonstration of the component-based approach, it will suffice. KST-based components and modules can always be exchanged by others at a later point.

The game to be designed is inspired by the afore-mentioned *Star Wars* RPG *Knights of the Old Republic*. In that game, robots ("droids" in *Star Wars* terminology) play an important role. Robots can join the PC's party and support him or her in his/her adventures, but also appear as adversaries. One can re-program robots, repair them, or boost their abilities through found equipment. In order to work on droids, the player needs a sufficiently good rating in the Repair skill and maybe a certain tool or piece of equipment. Programming droids in *Knights of the Old Republic* is not carried out in depth, i.e. the player need not type in algorithms or solve puzzles combining various control chips to effect a behaviour. Yet, he could.

Our educational science fiction RPG would initially play out in three settings. In the beginning, the PC enlists as a pilot or engineer in the game universe's spaceship fleet. His or her first quest is then to deliver goods to a far-away star system. To reach his destination, the player is to navigate his spaceship through an asteroid field. If he fails, the spaceship is destroyed. If he succeeds, the craft will still have received minor damage, forcing the PC to perform an emergency landing on the largest asteroid. As it turns out, the asteroid is a gigantic wasteyard planet. The player will find a large selection of half-working, half-broken equipment as well as a variety of robot parts. Some equipment is useful, some dangerous. In order to survive on this planet, and ultimately repair the ship or send out a mayday call, the PC will have to build, repair and re-program robots to accompany him.

This is where the game-based ITS can test and train steering algorithms. The spaceflight part tests and trains basic linear algebra as part of the navigation task. While enlisting in the fleet, the character is being created, which in our approach mean the same as: The player's knowledge is being assessed.

The rules system uses a range of character feats – Boolean-valued abilities, which the player and in the game his or her avatar can know or not know – related via prerequisite relationships. In KST, this is called the surmise relation, in our case:

$$\neg \textit{Navigate Using Vector Norm} \Rightarrow \neg \textit{Navigate Using Normed Vector}\,. \tag{1}$$
$$\neg \textit{Navigate Using Addition} \Rightarrow \neg \textit{Program Seek}\,. \tag{2}$$
$$\neg \textit{Navig. Using Scalar Multiplication} \Rightarrow \neg \textit{Program Seek}\,. \tag{3}$$
$$\neg \textit{Navigate Using Normed Vector} \Rightarrow \neg \textit{Program Seek}\,. \tag{4}$$
$$\neg \textit{Program Seek} \Rightarrow \neg \textit{Program Flee}\,. \tag{5}$$
$$\neg \textit{Program Seek} \Rightarrow \neg \textit{Program Arrive}\,. \tag{6}$$
$$\neg \textit{Program Seek} \Rightarrow \neg \textit{Program Pursuit}\,. \tag{7}$$
$$\neg \textit{Program Seek} \Rightarrow \neg \textit{Program Follow Closed Path}\,. \tag{8}$$
$$\neg \textit{Program Arrive} \Rightarrow \neg \textit{Program Follow Open Path}\,. \tag{9}$$
$$\neg \textit{Program Flee} \Rightarrow \neg \textit{Program Evade}\,. \tag{10}$$
$$\neg \textit{Program Pursuit} \Rightarrow \neg \textit{Program Evade}\,. \tag{11}$$
$$\neg \textit{Program Follow Closed Path} \Rightarrow \neg \textit{Program Follow Path}\,. \tag{12}$$
$$\neg \textit{Program Follow Open Path} \Rightarrow \neg \textit{Program Follow Path}\,. \tag{13}$$
$$\neg \textit{Program Pursuit} \Rightarrow \neg \textit{Program Offset Pursuit}\,. \tag{14}$$
$$\neg \textit{Program Offset Pursuit} \Rightarrow \neg \textit{Prog. Offs. Purs. with Arrive}\,. \tag{15}$$
$$\neg \textit{Program Arrive} \Rightarrow \neg \textit{Prog. Offs. Purs. with Arrive}\,. \tag{16}$$

For instance, surmise relationship 2 means: If the player cannot navigate the spaceship using vector addition, then we know for sure that the player cannot program a robot to seek, i.e. to walk towards a certain target spot, either. This is because the Seek formula requires vectors to be added. The ITS needs to teach the player how to add vectors first. Notice that the surmise relation is reflexive and transitive. For instance, $\neg$*Navigate Using Normed Vector* $\Rightarrow$ $\neg$*Program Arrive* is true as well.

Any valid set of known items is a possible knowledge state. A beginner might know $\{$*Navigate Using Addition*, *Navigate Using Scalar Multiplication*$\}$. Making use of the surmise relation, one can compute the knowledge space, i.e. the set of all possible knowledge states.

Besides information like the character's name, appearance, or equipment, the character sheet module of our RPG would list the set of the PC's known feats, which is equivalent to the player's knowledge state. When advancing in the game, the character would acquire more and more feats to reflect the player's learning. In this example, the pilot or engineer would learn new ways of setting a course through space, or new ways in which to program a robot.

The drama manager is the component that chooses what to teach next, i.e. the actual tutor.

Its quest manager module queries the character sheet and the rules system to determine desired next knowledge states, i.e. to find knowledge states that extend the current knowledge state by exactly one feat. That is, there are no further prerequisite relationships with which to deal first. To determine desired knowledge states, a data structure called the basis of the knowledge space can be applied [13]. The quest manager then generates quests for the feats that extend the knowledge space. For instance, the player might already know all the items related to vector maths. The spaceship navigation quest could be left out, and the player would start directly on the planet. As he cannot *Program Seek* yet, he or she would receive an appropriate quest.

Choosing quests for feats to be learnt may be a static one-to-one mapping using a database, or a dynamic generation process. Dynamic generation might create one quest based on several feats to be learnt. Or it might still base one quest on one feat, but dig deeper into knowledge items and see of which smaller units their corresponding tasks are composed. The database might store smaller, reusable units instead of whole quests. In the context of knowledge spaces, Albert and Held [1] analyse knowledge items and model their inner structures by task/problem "components" (not to be confused with our software components).

A quest story may be based on a known plot pattern. For instance, searching the junkyard planet on which our PC had crash-landed for a means of escape would be based on the "Robinson Crusoe" plot pattern. In [29], we collect such patterns, and model potential quest stories using Petri nets. For static quests, these Petri net structures might be used as a whole. Maybe *Program Seek* needs to be learnt to program a robot – by whatever means – to find a way of escaping, like a cave entrance leading to an ancient communication device. For dynamic quest generation, individual Petri net states could represent smaller-unit task components. Maybe, each task related to a feat can utilise a different selection of reusable pieces of equipment, e.g. control chips for programming the robots, that first must be found and then be assembled correctly. Mappings of this kind may or may not work with our plot patterns – this is a subject for future work.

Quests are given to the PC by non-player characters controlled by the NPC manager. In addition to handing out quests and accepting results, NPCs would further carry out the initial assesment and deliver feedback to the user. An NPC might be a commander of the spaceship fleet who initially employs the PC – the character creation process or assessment – and later can be contacted for help via the ship's radio. It is the overall NPC manager module though that knows the actual algorithms for assessment – in KST e.g. [14] – and giving feedback. Acting like a puppeteer, it will send out its NPCs to deliver its messages to the player and obtain input from him or her.

To determine whether a quest has been completed, the RPG-based ITS needs its environment manager. This is the simulation engine that carries out the player's actions. Having inserted control chips into a robot in a certain configuration, what will the robot do? It might walk into the wrong direction and hit a wall, which causes further damage to it. Or it might not walk at all. The Plug 'n Train framework is already interfaced with a simulation engine, *James II* [19].

The above modules work independently of the user interface. Only once the NPC manager knows what an NPC is to say or once the environment manager knows where the robot will walk or whether it will hit a wall, these will inform the user interface what to show on the screen. RPGs use a visual representation of the virtual world in which the story takes place. In our case, this would be an interactive 2D, isometric, or 3D visualisation of the junkyard planet and its inhabitants. Plug 'n Train is already interfaced with *Adobe Flash* which allows straightforward authoring of 2D game environments. An isometric perspective should be possible as well, since internally it could use a 2D representation. Using our component-based approach, we can start with simple graphics, and at some later point exchange the user interface module with another one employing real 3D. In the story outlined above, the biggest problem would be the spaceflight environment in which we would teach linear algebra by interactive spaceship navigation. It does not fit with the more typical planetside RPG environment, and it might require 3D from the start. On the other hand, one of the very first computer RPGs, *Ultima* (Richard Garriott, 1981; for its historical significance see [17]), used a 2D bird's eye view along with a 3D spaceflight interface.

Implementing a game like this would be the first step towards a generic game-based ITS architecture. By using the component-based approach, knowledge domain and story content could be exchanged with relative ease.

4 Discussion

For the design of adaptive digital educational games, the proposed architecture offers a number of advantages. The component-based approach – which uses exchangeable modules – and the genre-based approach – genres are held together by common, recurring principles – have the potential of improving reusability. In addition, the eLearning author is already given an architecture that is both game engine and ITS architecture. He or she can create a story of educational value without having to figure out how the integration would work on a technical level.

Regarding adaptivity, we believe that the architecture would support both non-invasive learner modelling [25] and scrutable learner modelling (e.g. [22]).

Pre-learning assessment in many traditional ITS is cumbersome. Character creation, on the other hand, is fun! It is totally natural for RPG players to state abilities, skills, and interests using precise numbers or a multiple-choice dialogue. Players who are less into this genre can stick to the class-selection or the foundling approach. These would correspond with ITS that reduce assessment to picking a level of experience. Therefore, in our genre approach, assessment would not invade or intrude into the fun of playing the game.

Traditional ITS may hide the user model. Yet, it should be better for conscious learning to be able to access one's knowledge state from time to time. RPG players are used to inspect their character sheets regularly, e.g. when levelling up and raising skills. It is part of the fun. We therefore believe that we have proposed a viable approach to scrutable user modelling as well.

5 Conclusions and Future Work

This paper introduced a component-based ITS architecture based on the genre of computer role-playing games. We believe that our proposal will help narrow the gap between playing and learning in digital educational games, so that one day these will go together seamlessly. In order to be able to validate our claims, one system including all components and modules is to be implemented.

We are going to detail each component further. This is necessary, as the proposed architecture, although being a software-engineering model, is still on a rather high level. Especially, the adaptivity part needs to be formalised. Where should we adapt – on a micro or on a macro level [25]? Even though our experienced learner might want to skip the part on linear algebra, he or she might not want to miss out on the spaceflight sequence. Maybe the whole story should be told, and rather the quests themselves should be adapted.

We will implement a first prototype (probably with a different first knowledge domain) using the existing Plug 'n Train framework, and then evaluate it both technically and with users. Development would begin with individual modules, and from there it would work towards a whole RPG-based intelligent tutoring system.

References

1. Albert, D., Held, T.: Component-based knowledge spaces in problem solving and inductive reasoning. In Albert, D., Lukas, J. (eds.): Knowledge Spaces: Theory, Empirical Research, and Applications, Lawrence Erlbaum Associates, Mahwah (1999).

2. Bormann, M., Heyligers, K., Kerres, M., Niesenhaus, J.: Spielend Lernen! Spielend Lernen? Eine empirische Annäherung an die Möglichkeit einer Synthese von Spielen und Lernen. In Lucke, U., Kindsmüller, M.C., Fischer, S., Herczeg, M., Seehusen, S. (eds.): Workshop-Proceedings der Tagungen Mensch und Computer 2008, DeLFI 2008 und Cognitive Design 2008, Lübeck, Germany, September 2008. GI, logos Verlag, Berlin (2008) 349–354 (in German).
3. Buckland, M.: Programming Game AI by Example. Wordware, Plano (2005).
4. Burgos, D., Moreno-Ger, P., Sierra, J.L., Fernández-Manjón, B., Specht, M., Koper, R.: Building adaptive game-based learning resources: The integration of IMS Learning Design and <e-Adventure>. Simulation & Gaming 39(3) (2008) 414–431.
5. Burton, R.R., Brown, J.S.: An investigation of computer coaching for informal learning activities. In Sleeman, D., Brown, J.S. (eds.): Intelligent Tutoring Systems. Academic Press, London (1982) 79–98.
6. Carbonaro, M., Cutumisu, M., Duff, H., Gillis, S., Onuczko, C., Schaeffer, J., Schumacher, A., Siegel, J., Szafron, D., Waugh, K.: Adapting a commercial role-playing game for educational computer game production. In 2nd International North American Simulation and AI in Computer Games Conference (GAME'ON-NA'2006), Monterey, USA, 19–20 September 2006, 54–61.
7. Carro, R.M., Breda, A.M., Castillo, G., Bajuelos, A.L.: A methodology for developing adaptive educational-game environments. In De Bra, P., Brusilovsky, P., Conejo, R. (eds.): Adaptive Hypermedia and Adaptive Web-Based Systems – Proceedings of the Second International Conference on Adaptive Hypermedia (AH 2002), Malaga, Spain, 29–31 May 2002. Springer Verlag, Berlin/Heidelberg (2002) 90–99.
8. Clancey, W. J., Kintsch, W.: Methodology for building an intelligent tutoring system. In Miller, J. R., Polson, P. G. (eds.): Methods and Tactics in Cognitive Science, Lawrence Erlbaum Associates (1984) 51–84.
9. Conati, C., Zhao, X.: Building and evaluating an intelligent pedaogical agent to improve the effectiveness of an educational game. In IUI/CADUI 2004, Madeira, Portugal, 13–16 January 2004.
10. Crawford, C.: Chris Crawford on Interactive Storytelling. New Riders Games, Berkeley (2005).
11. Doignon, J.-P., Falmagne, J.-C.: Spaces for the assessment of knowledge. International Journal of Man-Machine Studies 23 (1985) 175–196.
12. Doignon, J.-P., Falmagne, J.-C.: Knowledge Spaces. Springer-Verlag, Berlin (1999).
13. Dowling, C.E.: Applying the basis of a knowledge space for controlling the questioning of an expert. Journal of Mathematical Psychology 37(1) (1993) 21–48.
14. Dowling, C.E., Hockemeyer, C.: Automata for the assessment of knowledge. IEEE Transactions on Knowledge and Data Engineering 13(3) (2001) 451–461.
15. Fritz, J.: Das Spiel verstehen. Eine Einführung in Theorie und Bedeutung. Juventa (2004) (in German).
16. Gómez-Martín, M.A., Gómez-Martín, P.P., González-Calero, P.A.: Game-driven intelligent tutoring systems. In Proceedings of the Third International Conference on Entertainment Computing, Eindhoven, The Netherlands, 1–3 September 2004. Springer Verlag, Berlin/Heidelberg (2004) 108–113.
17. Hallford, N., Hallford, J. Swords & Circuitry: A Designer's Guide to Computer Role-Playing Games. Prima Tech, Roseville (2001).
18. Harr, R., Buch, T., Hanghøj, T.: Exploring the discrepancy between educational goals and educational game design. In Proceedings of ECGBL 2008 – The 2nd European Conference on Games Based Learning, Barcelona, Spain, 16–17 October 2008, 165–174.

19. Himmelspach, J., Uhrmacher, A.: Plug 'n simulate. In Proceedings of the 40th Annual Simulation Symposium, IEEE Computer Society (2007) 137–143
20. Huizinga, J.: Homo ludens. Vom Ursprung der Kultur im Spiel. 19. Auflage. Rowohlt (2004) (in German).
21. Jantke, K.P.: Serious Games - eine kritische Analyse. In 11. Workshop M̈ultimedia in Bildung und Wirtschaft¨- Tagungsband, Ilmenau, Germany, 20–21 September 2007, 7–14 (in German).
22. Kay, J., Li, L.: Scrutable learner modelling and learner reflection in student self-assessment. In Intelligent Tutoring Systems, Springer Verlag, Berlin/Heidelberg (2006) 763–765.
23. Katzlinger, E.: Online Plan-Rollenspiel in der E-Business Ausbildung. In Lucke, U., Kindsmüller, M.C., Fischer, S., Herczeg, M., Seehusen, S. (eds.): Workshop-Proceedings der Tagungen Mensch und Computer 2008, DeLFI 2008 und Cognitive Design 2008, Lübeck, Germany, September 2008. GI, logos Verlag, Berlin (2008) 349–354 (in German).
24. Khan, M.M.: Implementing an intelligent tutoring system for adventure learning. The Electronic Library 20(2) (2002) 134–142.
25. Kickmeier-Rust, M.D., Göbel, S., Albert, D.: 80Days: Melding adaptive educational technology and adaptive and interactive storytelling in digital educational games. In Klamma, R., Sharda, N., Fernández-Manjón, B., Kosch, H., Spaniol, M. (eds.): Proceedings of the First International Workshop on Story-Telling and Educational Games (STEG '08) – The Power of Narration and Imagination in Technology Enhanced Learning, Maastricht, The Netherlands, 18–19 September 2008.
26. Koubek, J.: Lernen wie im Spiel – Kompetenzmodelle von Unternehmen und Rollenspielen im kritischen Vergleich. In Sieck, J., Herzog, M.A. (eds.): Kultur und Informatik: Serious Games, Berlin, Germany, 14 May 2009. Verlag Werner Hülsbusch, Boizenburg (2009) 89–114 (in German).
27. Lelouche, R.: Intelligent tutoring systems from birth to now. KI Zeitschrift 4/99 (1999) 5–11.
28. Maciuszek, D., Martens, A.: Virtuelle Labore als Simulationsspiele. In Workshop Pervasive University (PerU 2009) as part of 39. Jahrestagung der Gesellschaft für Informatik (Informatik 2009), Lübeck, Germany, 29 September 2009 (to appear) (in German).
29. Maciuszek, D., Martens, A.: Story patterns for the design of game-based learning experiences. In 3rd European Conference on Games Based Learning (ECGBL 2009), Graz, Austria, 12–13 October 2009 (to appear).
30. Martens, A.: Discussing the ITS Architecture. In Proceedings of the GI Workshop Expressive Media and Intelligent Tools for Learning, 26th German Conference on Artificial Intelligence KI-2003, Springer Verlag, Berlin/Heidelberg (2003) 396ff.
31. Mills, C., Dalgarno, B.: A conceptual model for game based intelligent tutoring systems. In ICT: Providing Choices for Learners and Learning, Proceedings ascilite, Singapore, 2–5 December 2007.
32. Oertel, M., Himmelspach, J., Martens, A. Teaching and training system plus modeling and simulation – a plugin based approach. In Proceedings of the EUROSIM/UKSim, IEEE, IEEE Computer Society Conference Publications, CPS (2008) 475–480.
33. Peirce, N., Conlan, O., Wade, V.: Adaptive educational games: Providing non-invasive personalised learning experiences. In Eisenberg, M., Kinshuk, Chang, M., McGreal, R. (eds.): 2nd IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning, Banff, Canada, 17–19 November 2008, 28–35.

34. Reynolds, C. W.: Flocks, herds, and schools: A distributed behavioral model. Computer Graphics 21(4) (SIGGRAPH '87 Conference Proceedings) (1987) 25–34.
35. Rickel, J.: Intelligent computer aided instruction: A survey organized around system components. In IEEE Transactions on Systems, Man, and Cybernetics 194(1) (1989) 40–57.
36. Riedl, M., Lane, H.C., Hill, R., Swartout, W.: Automated story direction and intelligent tutoring: Towards a unifying architecture. In Papers of the Narrative Learning Environments Workshop at the 12th International Conference on Artificial Intelligence (AIED05), Amsterdam, The Netherlands, 18–22 July 2005.
37. Scheuerl, H.: Das Spiel. Untersuchungen über sein Wesen, seine pädagogischen Möglichkeiten und Grenzen. Beltz (1994) (in German).
38. Siemer, J., Angelides, M.C.: Embedding an intelligent tutoring system in a business gaming-simulation environment. In Tew, J.D., Manivannan, S., Sadowski, D.A., Seila, A.F. (eds.): Proceedings of the 1994 Winter Simulation Conference, ACM, Lake Buena Vista, USA, 11–14 December 2004.
39. Spring, F., Ito, T.: Combining personalisation and adaptation in game-based learning systems. In IFIP WG 3.1, 3.3 & 3.5 Joint Conference, Ålesund, Norway (2006).

Adapting Contents and Procedures in Educational Video Games with Collaborative Activities

Natalia Padilla Zea, Francisco L. Gutiérrez, Nuria Medina Medina, José Luís González Sánchez

Video Games and E-Learning Research Group. Research Group GEDES. University of Granada, Spain.
C/ Periodista Daniel Saucedo Aranda s/n, 18071, Granada.
{npadilla, fgutierr, nmedina, joseluisgs}@ugr.es

Abstract. Learning processes are difficult tasks that children must face along their whole scholar life. It is proved enough that there are several techniques that make these processes easier either regarding learning styles or techniques itself. One of the most popular techniques is collaborative learning, which we include into educational video games in order to do the learning process more attractive. However, also this technique can be adapted according to students' preferences or skills to have as much success as possible. So, in this paper we present a set of models that we use to analyze the playing / learning process and to adapt several aspects in the game which help students to acquire more curricular contents.

1 Introduction

In recent years, New Technologies have been incorporated into our everyday life. We have been witness of this process and we have tried to adapt our activities to this new way of working. However, children have already grown in a "virtual world": they do not know cassettes or maps and they are used to see their parents reading their emails. Even, New Technologies have been incorporated to schools. Nobody is already surprised when sees a child learning to read or write by using a computational tool. Actually, it is proved that by incorporating these technologies into pupils' educational development, it is possible to improve important aspects as: their cognitive skills, the time that they spend learning, their motivation to learn, their concentration and their attention [1].

As a particular case of New Technologies, we can highlight video games. Several studies have proved that learning can be achieved by playing. Moreover, there are several lines of research that claim for the use of video games as learning tools, due to they become the learning time into a funny process, what adds many advantages to the children's attention and motivation [1], [2], [3].

Our main interest is focused on including collaborative activities into educational video games in order to do learning processes more interesting since Computer-Supported Collaborative Learning (CSCL) is a well-known discipline which improves learning capability, favors that pupils learn better from interaction with other group members and get attitudes closer [4].

Another quality that video games have is that they are highly interactive mediums and, from a technological perspective, they are an ideal medium to support an adaptive learning experience. Actually, we can develop video games that can monitor players' performance and change some parameter of working during the match [5].

Adaptations can be made regarding two aspects. By one hand, at educational level we can favor learning processes and personalize them according to students' profiles. By another hand, at video game level we can analyze playability from each of players' standpoint and adapt the video game in order to maintain an optimal degree of playability. By doing this, players will be attentive to the playing process (and implicitly, to the learning process).

To do these adaptations in an effective way, a customized monitoring is needed. Usually, teachers cannot take attention to all activities and all students due to the number of students in a classroom and extra-tasks carried out at home. So, we think that this problem can be solved by using models. Specifically, we need a model to monitor what students do during a class at school and how they progress along a period of time.

The remainder of this article is organized as follows: Section 2 presents the elements and models needed to carry out an analysis and an adaptive process. In section 3 we explain how these models are interrelated. Section 4 talks about adaptations that we propose to apply at educational level and section 5 presents the need to adapt at video game level too. Finally, section 6 shows some conclusions and further work.

2 Models to adapt

In order to do adaptations in an educational video game with collaborative activities, we have designed a set of models which allow us to analyze several aspects of learning process. Our main interest is focused on the collaborative process, but we can study others aspects in the learning process by using these models.

We think that in a learning process by means of video games we must adapt as contents as procedures in order to do the learning easier and more effective to students. So, we have focused on three main aspects: goals, tasks and interaction, and we explain them in the next subsection.

According to these objectives of adaptation, we have four models to store and analyze the students' performance and to take decisions during and after the game. We explain these models in section 2.2.

2.1 Goals, Tasks and Interaction

Goals, tasks and interaction are three main elements in our model process. We need to define and characterize them in order to choose the best options to each match.

Goals

Goals that a student (player) must face during the learning process can be educational or recreational and both of them are related since recreational goals contribute to achieve educational goals. Both set of goals are dynamic because we can adapt them regarding to the achievements of students along a match.

According to the number of people that faces a goal, we have two types: individual and group. Individual goals must be achieved by each of players without be helped; Group goals must be achieved by the group commonly. Group goals favor interdependence between group members.

By other hand, group goals can be classified according to the number of players that can achieve it each time. So, we have competitive and non-competitive goals. Competitive goals are those that can be achieved by one player each time. So, when a player wins, the rest of players lose. Usually, this kind of goals is not used to be presented at educational level, but they are especially important at recreational level because they suppose an additional motivation. Non-competitive goals can be achieved by all players, independently of some players pass it before than others.

Individual and Group goals can be assigned by the teacher or by players (students) themselves. If goals are assigned by the teacher, he takes into account which educational goals students must practice and he chooses those recreational goals related to them. Otherwise, students organize their own learning and, in this way, they favor their planning and taking decisions skills. Similarly, tasks can be assigned by teacher or students, but in this paper we focus only on the first option.

Tasks

To achieve the goals, players (students) must carry out a set of tasks. In some cases, to achieve a goal, students can carry out different subsets of tasks. To complete a task, students must carry out a set of activities, which have a set of resources that can be used to do it. Tasks can be individual or group, and we distinguish three types in group activities: 1) *Simultaneous*: All group (or sub-group) members working in the task have to face each activity at the same time. This kind of tasks favors collaboration inside the group and, in an indirect way, individual learning for each group member; 2) *Ordered task*: To achieve the task some group members must take part. It is not necessary to do it at the same time, but activities must be done in a specific order. So, to do an activity, the previous ones must be done before. Constraints about the sequence of activities can only affect some of the activities in a task, and not the others. In any case, this way of facing tasks favors interdependence between group members because they are aware of they need their partners to achieve the task; 3) *Non-ordered task*: Each player can carry out his part (subset of activities) when he wants and it does not have any influence in the rest of the group members' work. To carry out a task they only have to do the corresponding activities. This kind of tasks, in a similar way that ordered tasks, improve cooperative attitudes since group members face the task as a team, but they have responsibilities only in a part of it.

Interaction

During a collaborative learning process by means of video games, three types of interactions can appear: communication, collaboration and coordination. We think that to detect and to classify messages occurring between group members while they are interacting is a basic element in the analysis process and can help us during the adaptive process. In order to do this classification easier, we have done a strong messages categorization [6], using three categories: 1) *Communication Messages*: They are messages whose intention is to inform about something. For example, in this category we have: question and answers, sharing information, checking how the partners are working or social messages. Social messages are those which are not related to the game, but inform us about relationships existing inside the group; 2) *Collaboration Messages*: They take place during a collaborative situation, such as pose a proposal, ask for help, ask for resources, propose an idea or vote a situation; 3) *Coordination Messages*: We use this kind of messages when we organize methods or strategies to achieve a goal. Taking decisions, negotiating a solution or planning tasks are included in this category.

Information obtained from these three elements (goals, tasks and interaction) we think that can be enough to analyze collaboration and learning processes occurred during the video game. In this way, we can act on them if necessary.

2.2 Modeling to Analyze and Adapt

Our proposal is to store all relevant information to be able to do adaptations during the learning process. We have created some models in order to organize information in an useful way. In particular, we define four interconnected and interrelated models: Tasks and Goals Model, State of the Game Model, Player Model and Group Model.

Tasks and Goals Model

Since educative videogames try to achieve an implicit learning (without the users be aware of their learning process), we consider that it is necessary to separately define didactic goals and recreational goals. However, both types of goals must be connected, so that when the user reaches a goal of the videogame he also achieves the didactic goal or goals related to that satisfied recreational goal.

With this propose we establish two levels in the Tasks and Goals Model: 1) The Educative Level (inferior layer in Fig. 1) and 2) The Video Game Level (superior layer in Fig. 1). We have named as V- to video game's tasks and E- to educational tasks. After V and E we include a sequence number. If the name of task fits in the model, we can write it instead of a generic name. The relation among the tasks of both levels is named "implementation". So, when a task Vx is useful to teach the task Ey, we say that Vx implements Ey. In a graphical representation it is noted as a discontinue line. In Fig. 1, V1 implements E1, V3 implements E2, and V4 implements E2 and E3. As we can see in that figure, V2 does not implement any task at educational level. That is because video games can have some tasks which are not related to an educational content, but they are only for fun.

Let's think in a specific video game and see Fig. 1. Educational-Goal 2 is multiply by 2. From an educational standpoint, we want that students learn to add 2+2, the result added plus 2, and so on (E3). By other hand, we want that students memorize multiplication table (E2). At video game level, Game-Goal 2 is to come into a room, but stairs are broken. So, we have to build a stair to go up. First of all, the player has to build the stair by using a set of disordered blocks near the building. Because the avatar has only two hands, he has to take one block in each hand. Each step must have two more blocks than the previous one and the stair must have twenty steps (V3). When the player arrives to the door, a password is needed (V4). The player will have to answer a result of a particular item from the multiplication table.

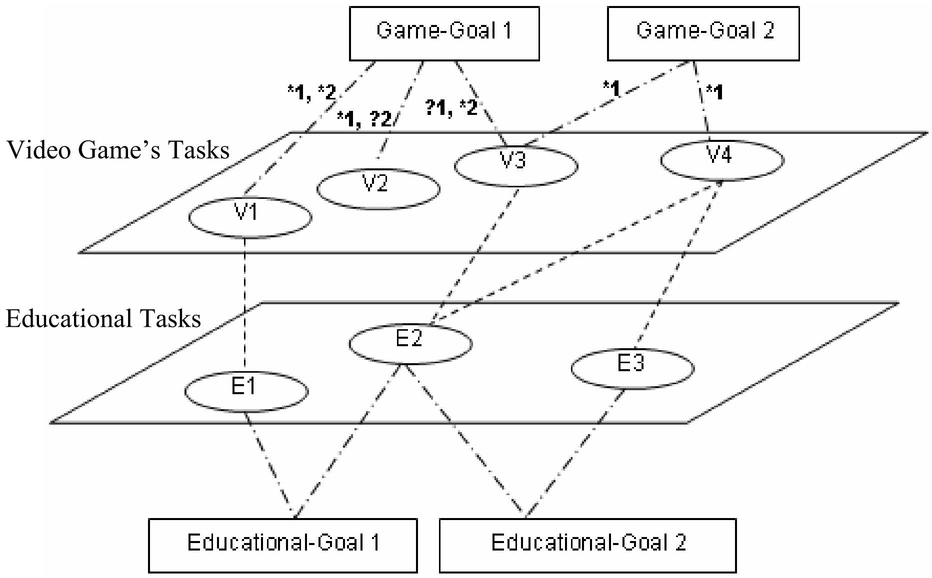


Fig 1. Tasks and Goals Model.

By other hand, each task at video game level can satisfy one or more objectives of the videogame. Similarly, a task at educational level can satisfy one or more learning objectives. Although it cannot be observed in the example in Fig. 1, in both levels, the goals are organized in a hierarchy. The hierarchy allows us to structure a goal in sub-goals which are directly associated with the tasks. The same sub-goal can be associated with several goals. In addition, the tasks associated with a goal can be obligatory or optional to accomplish that goal. So, the players must perform all the obligatory tasks but not necessarily the optional tasks, which contribute additional aspects. In the model we use the symbol * to represent obligatory tasks and the symbol ? to indicate voluntary tasks. Due to a task can be associated to one or more goals, the obligatorily / optionally symbol must be written on each association, not on the task. Thus, a task can be obligatory for a goal but optional for another one. In addition, different ways of accomplish a goal can be established. Each path is defined as a specific subset of tasks that must be performed and each subset of tasks is identified by a sequential number. For example, in order to satisfy the goal Game-Goal 1, we can complete the obligatory tasks V1 and V2, or V1 and V3. For Game-Goal 2, both V3 and V4 are obligatory. Optionally, players can carry out the third one. The intersection of the subsets of tasks cannot be empty, so an obligatorily / optionally symbol can be labeled with several numbers separated by colons.

Moreover, another aspect related to the Tasks and Goals Model is the set of tools that can be utilized to perform the tasks. Basically, a player, while executes a task, can need two kinds of tools: collaborative tools (chat, forum, discussion room, group stock, etc.) and game tools (a map of the virtual world, a key, a notebook, etc.). Each task has associated a set of tools, so it is possible to detect if the players have used all the available tools. In negative case, the teacher can promote the use of certain tools in the next games (for example, the discussion room makes that the students clarify their questions more easily, and in addition allows them to reach a consensus and extend the learned concepts).

In order to assign goals and tasks properly, we have defined two attributes for each goal: *experience* and *difficulty*. In this model, experience indicates the minimum level of experience that a player should have to face this goal and difficulty suggests the initial difficulty level recommended to this goal. Similarly, tasks have attributes for experience and difficulty: Experience indicates the recommended level of experience for this task and difficulty informs about the optimal level for this task. A goal should have tasks with different level of experience and difficulty in order to allow the teacher to assign tasks according to the students' needs.

These two attributes can be updated during the game based on parameters as:

- A student needs to repeat a task
- A student is bored because he achieves secondary or not many interesting goals.
- Experience acquired by a student in other tasks or in previous games
- How far a student is to achieve a particular goal.

State of the Game Model

The State of the Game Model stores information about the current game. We distinguish two parts in this model: individual players' area and group's area (Fig. 2). This model is destroyed when the game finishes, therefore, it is a short-term model [7] which temporal life is limited to the duration of each the game.

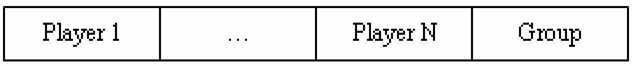


Fig 2. State of the Game Model

Table 1. Elements registered for each task

Element	Considerations	Main educational use
Both Player's and Group's Areas		
Task's ID	Extracted from Task and Goal Model. If group task, register type (non-ordered, ordered or simultaneous). Register experience and difficulty at this moment.	Know which educational aspect is been learnt.
Game Point	Phase or level in the video game in which a task is being carried out.	How well they are playing / learning.
Pursued Goal	Goal that students want achieve by carrying out this task.	Which educational goal is working on.
Beginning	Date and time when the task begins.	Delimits how many time last the task
End	Date and time when the task finishes.	To calculate how much time the task has lasted.
Activities	For each activity in the task, we store: beginning, end, number of failures and obtained score.	To know which activities are more difficult.
Events	We define some interesting events to analyze. We store: type, beginning, end and specific data	Depending on the event.
Group's Area		
Members working	Group members working on a task. Subgroup working on each particular activity.	To identify degree of participation of each student.
Members interacting	We classify them by activities. For each message: Date, time, communication tool, sender, recipient and content of the message.	To track coordination and planning processes.

Information maintained by this model is an events' log. The log registers events that we have defined as interesting, that is, all the significant events in the game. These events will be analyzed later to identify the relevant information about the players, the group and the context in which tasks are executed.

The group's area registers information concerning to tasks that the group performs together. The individual players' area keeps the information concerning tasks performed by each player alone. In Table 1 we can see elements registered for each task.

Player and Group Models

They are long-term models [7] since their information persists during the whole learning process and it is updated with relevant information from the analysis of State of the Game Model. Since in this context a group is more than the addition of its parts, we need a player model for each of players in the group and a model for the own group.

For the assignment of goals, we need to know some information about players and group. This information can be looked up in a profile included in the Player and Group Models. This profile includes:

- Player's / Group's identification.
- If a group, a list of identifications of group's players.
- Type of activities in which player / group obtains best scores (learn more). We have this information in order to detect which type of activities the player solves more easily. When the score does not increase, we can assign one of these activities to avoid that the player gives up.
- Type of activities in which player / group obtains worst scores (learn less). If teacher notes that a player does not improve their knowledge, he can avoid this kind of activities. By other hand, if a player always leaves these activities, the teacher or the system can modify them and put this activity as mandatory, or to include more activities of this type in the proposed tasks for this player.
- Number of time that player / group has played the video game.
- Total number of hours of playing.

Regarding to the Group Model and based on group learning methods, we can distinguish two kinds of groups:

- Formal Groups: Students are grouped permanently for a long period of time, for example, a semester.
- Temporal Groups: Students are grouped for specific activities but this group is not last in the time, it is said, the group disappears when the activity finished.

Only for formal groups a long-term group model is needed. In this case, group's information is updated after a match by analyzing information in the group zone of State of Game Model. So, the Group Model exists while the corresponding group exists in the learning process. For temporal groups, information is used to complete some aspects in the Player Models of players belonging to this group, but a Group Model is not created. Specific elements in Player and Group Models can be sawn in Table 2.

Table 2. Elements in Player and Group Models

Element	Considerations	Main educational use
	Both Player and Group Models	
Proposed Educational Goal	Teacher can assign educational goals to solve in one or more matches. Teacher assigns also values for experience and difficulty. Experience and difficulty must according to player's or group's profile.	To adapt educational contents to needs of each student.
Proposed Educational Tasks	Specific tasks for each goal. Values for experience and difficulty are needed.	Adapt goals achievement to students needs.
Passed Educational Tasks	Educational tasks passed properly by players in previous matches. Values for experience and difficulty with which task has been passed is also stored.	Aspects of a task that students have overcome and in which level.
Passed Educational Goals	Goals that players in previous matches. How well goals have been achieved is related to corresponding tasks. Values for experience and difficulty are also needed.	Goals that students have passed.
	Group Model	
Most prestigious node	Member who receives a greatest number of messages.	Related to personal accountability [8] to share knowledge.
Most influential node	The player who sends the greatest number of messages. Distributes more information than his partners.	Related to personal accountability [8] to share knowledge.
Network's density	Relation between existing and possible links.	Are all group members related? How many interactions occur inside the group?
Adjacency matrixes	How many messages have been sent for each type and subtype of message.	Obtain SNA indexes and graphical representations to detect problems.

We want to highlight specific information in the Group Model. Here, we store information about interaction occurring during group activities development. From this information we can build a set of networks in order to study interactions inside a group. To build and to analyze these networks, we use formal methods from Social Network Analysis (SNA). In this kind of networks, a node represents a member of the group and a link between two nodes represents an interaction between these two particular nodes. Next, we explain the indexes that we extract from a network by using SNA in order to do our knowledge of interactions deeper.

Regarding to collaboration analysis, to create social networks to represent interaction occurred during a learning process allow us to act on students groups when we detect some problems. Examples of social kind problems can be:

- Some group members depends on another excessively
- A member does not take part enough
- Low level of responsibilities sharing.
- There are too many or two little communication in the group

To detect some of these problems can provoke adaptive activities during the playing / learning process.

3 Integrating Models

The main objective of the proposed models is to analyze collaboration to do adaptations during and after the game. At the beginning of the game, the teacher assigns goals and tasks to players and groups and this information goes to the Player and Group Models (Figure 3, arrows A and B). During the game, the State of the Game Model registers actions performed by players and stores them in the player and group zone, depending on the type of activity. The State of the Game Model remains the whole match. During this time, we can make analysis of the information in order to adapt some aspect of the game, apart from the final analysis, when the match is over. After the match, the whole State of the Game Model is analyzed and the information extracted from it is stored in the Player and Group Models.

In this way, information transferred from the State of the Game Model (players' zone) to Player Models (Fig 3, arrow F1) outlines significantly how actors have performed during the match. If a player has solved several tasks regarding the same educational content, the mark stored for this educational content is the average mark of all related tasks. The mark for a task is a lineal combination of the number of mistakes, the score awarded and the time spent in that task. Since we have a historical record, the teacher can observe the player's evolution along the time.

In a similar way, information in the State of the Game (group's zone) goes to the Group Model (Fig 3, arrow F2). Apart from transformations cited before, now we have to analyze messages stored in the log in order to classify them according to the message categorization and to build the adjacency matrixes in each level. Also, we have to calculate indexes in each network in a match.

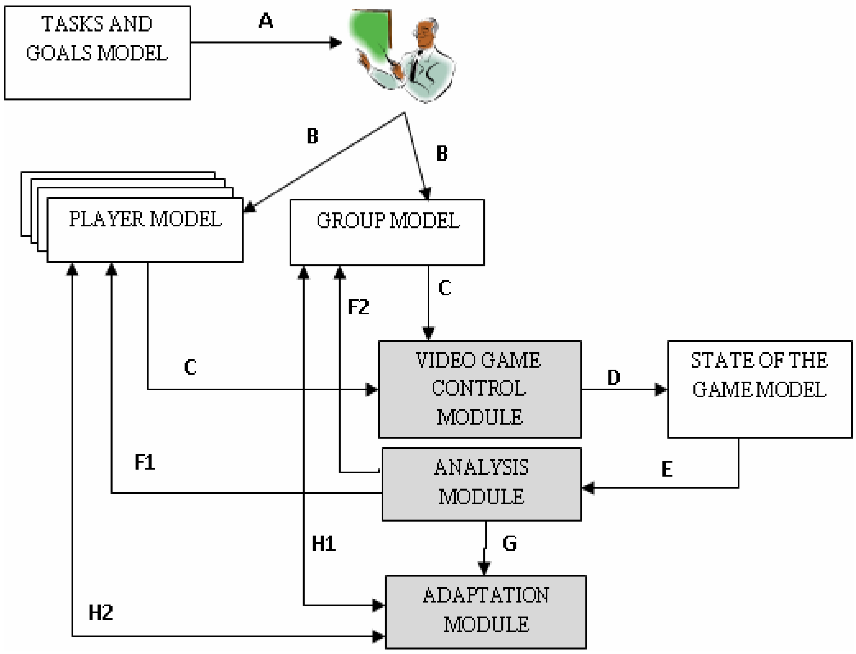


Fig 3. Model Integration for the Analysis and Adaptation Process

The Video Game Control Module has some responsibilities, but in this paper we want to remark that it checks that tasks faced by players are related to goals assigned by teacher (Fig 3, arrow C).

4 Analyzing Performance and Adapting

We must not forget that this set of models has a main objective: To obtain a global vision of the learning process and to act in order to do it better. In general, once we have this information, we can decide between one of these options:

– *To do nothing*: Once we have the results, we present them to the teacher without adding anymore.

– *To send alerts to the teacher*: We add some pieces of advices to the teacher in order to modify some level of difficulty or some tasks, or to make some interactions better.

– *To send alerts to the students*: Students / players receive some suggestions about the correct way to the solution or a clue to solve the problem. Regarding to interactions, students / players can be advised about the partner or partners who can help him. Obviously, these recommendations will be intended to modify unwanted relations or to foster relations with other group members.

– *To interact with students by means of a "fictitious" player or assistant*: The video game can incorporate some fictitious player or a wizard who gives advices to players when they request them or guide them when they have problems to solve anything.

Models defined are intended to store and analyze all relevant information and to summarize it to extract conclusions. So, when the Analysis Module takes information from the State of the Game Model (Fig 3, arrow E) it has to send information to Player and Group Models to register data in permanent models (Fig 3, arrows F1 and F2). In addition, it sends an alert to the Adaptation Module (Fig 3, arrow G) to inform it about a new analysis and it acts according to the selected option: to do nothing, to send alerts to the teacher or students or to interact by means of a fictitious player. Any adaptation that the system or teacher does must take information from the Player or Group Models and modify there the updated parameters (Fig 3, arrows H1 and H2).

We have classified several adaptations that can be done:

– *Planning of goals and tasks*: This type of adaptations can be done either for individual or group tasks and goals.
  - To propose the same task or goal but with a different level of difficulty. We can distinguish two cases: 1) if players pass goals / tasks more quickly than it is estimated or with very high score, *difficulty* must increase. 2) If players last too much to pass a goal / task or their score is two slow, *difficulty* must decrease in order to avoid boredom and to do the learning process more effective. This kind of adaptation is especially meaningful in group goals/tasks because it can reinforce or force some social skills. For example, if there is a "most influent node", we can assign him tasks with more coordination activities; or we can avoid it to encourage another kind of skills. These adaptations can be done automatically.
  - To change the subset of tasks to achieve a goal. As we said in section 2.2, there are several sets of tasks to achieve a goal. If teacher have assigned a particular set of tasks to a player or a group and he is not able to pass it, the teacher can modify this assignment and change activities according to the "Type of activities in which player / group obtains best scores". In this way, we make the game more interesting and help players to pass the challenges. By using this kind of adaptation we modify the process to learn a concept and we do our video game more flexible because it can teach children with different learning styles. This adaptation can be done automatic or semi-automatically.
  - To become optional tasks into obligatory tasks. If we realized that all activities of one particular type are not passed or passed with low score, we can do mandatory this kind of activities for a player or a group. By doing it, we can help students to gain skills in this kind of activities. As the previous one, this adaptation can be done automatic or semi-automatically.
  - Propose different goals. This adaptation must be done by a teacher. So, these results are informative. If a player / group have passed all sub-goals for a main goal, teacher can choose to change to another new goal, if scores and times are properly. Similarly, if there is a player or a group that cannot

achieve any goal, maybe a good option could be change to another goal and come to the first one back in another time.

– *Group structure*: If the Analysis Module has detected that some interaction problems arise, it could be a good solution to change some member of the group. We have two main strategies in order to improve collaboration into a group: to include a new member and to eliminate a member. For example, we include a member with high "influence" level if we need a member who coordinates activities or there are few interactions. Or we can eliminate a very "prestigious" member if the rest of group members only send messages to him, in order to improve relations between other members of the group.

5 Maintaining the playing process

We have talked about the learning process as a consequence of playing process. As we said in previous sections, one of the keys of learning by video games is that students become players and their attention and motivation are maintained during the whole process (or at least, more time than in a traditional class). So, we must not forget that video game must be also adapted to students' preferences.

To assess how funny or attractive a video game is can be a difficult issue. So, we have defined playability [9] in order to evaluate the player experience regarding to the own video game. In this way, we can adapt some aspects to improve the player's attraction and to maintain attention and motivation due to a funny activity. Playability is defined by a set of attributes and each of these attributes is assessed by some metrics [9]. We can see it in Table 3.

Table 3. Playability's attributes and metrics

	Metric name	Purpose
Effectiveness	Goal effectiveness	What proportion of the goals is achieved correctly?
	Goal completion	What proportion of the goals is completed?
	Attempt frequency	What is the frequency of attempts?
Efficiency	Goal time	How long does it take to complete a goal?
	Goal efficiency	How efficient are the users?
	Relative user efficiency	How efficient is a player compared to an expert?
Flexibility	Accessibility	What proportion of the goals can be achieved by using alternative ways of interaction?
	Personalization	What proportion of the personalization options are used by the players?
Safety	User health and safety	What is the incidence of health problems among users of the product?
	Software damage	What is the incidence of software corruption?

Satisfaction	Satisfaction scale	How satisfied is the player?
	Satisfaction questionnaire	How satisfied is the user with specific software features?
	Discretionary usage	What proportion of potential users chooses to use the system?
	Socialization	What proportion of potential users chooses to use the system?

These metrics are represented in the player's profile in order to update preferences regarding video games and to adapt several aspects of each particular video game to obtain advantages from this kind of learning.

By other hand, we need to evaluate difficult and experience al video game level, in a similar way we did at educational level. So, we have defined video game difficulty and video game experience in the Tasks and Goals Model and in Player and Group Models. This attributes have the same aim that their equivalent regarding to educational contents and they work similarly.

6 Conclusions and future work

In this paper we have outlined problems that teachers have to observe and analyze all activities for all students in their classes. By other hand, we have highlight how video games can be an excellent educational tool, in particular if they include collaborative activities. So, we have proposed a set of models to monitor each of students when they are learning by means of an educational video game.

In addition, we have proposed a set of parameters to do automatic, semi-automatic and manual adaptations along the match. These adaptations can be done at educational or video game level in order to do the learning process more effective.

Nowadays, we are working in the set of adaptations and parameters needed in the particular case of group activities. We want to implement some automatic actions to avoid that players realized that teacher is acting in the video game.

Acknowledgements This research is supported by the Spanish MCYT R+D projects TIN2008-06596-C02-02, TIN2007-64718 and TIN2007-60199 and the FPU Programme, and by the Andalusian Government R+D project P08-TIC-03717.

References

1. Nussbaum, M., Rosas, R., Rodríguez, P, Sun. Y., Valdivia, V. Diseño, desarrollo y evaluación de video juegos portátiles educativos y autorregulados. Ciencia al día Nº 3(2) (1999) pp: 1-20.
2. Gross, B. La dimensión socioeducativa de los videojuegos Revista electrónica de Tecnología educativa Nº 12 (2000).
3. McFarlane, A., Sparrowhawk, A., Heald, Y. Report on the educational use of games: An exploration by TEEM of the contribution which games can make to the education process. Cambridge, U.K.: TEEM (Teachers Evaluating Educational Multimedia) (2002).

4. Jong, B., Chan, T., Wu, Y., Lin, T. Applying the adaptive learning material producing strategy to group learning", International Conference Edutainment. LNCS, Vol. 3942 (2006) pp: 39-49.
5. Moreno-Ger, P., Sancho Thomas, P., Martínez-Ortiz, I., Sierra, J. L., Fernández-Manjón, B. Adaptive Units of Learning and Educational Videogames. Journal of Interactive Media in Education 2007/5 (2007)
6. Padilla Zea, N., González Sánchez, J. L., Gutiérrez, F. L., Cabrera, M. J., Paderewski, P.. From CSCL to VGSCL: A new approximation to Collaborative Learning. En: 1st International Conference on Computer Supported Education (CSEDU) (2009) 329-334.
7. Rich, E. Users are individuals: individualizing user models. Int. J. Man-Machine Studies 18, (1983) 199-214.
8. Johnson, D. W., Johnson, R. T. Learning Together. Sharan S. (Ed.), Handbook of Cooperative Learning Methods, Greenwood Press, Connecticut. (1994).
9. González Sánchez, J. L., Cabrera, M. C., Gutiérrez, F. L., Padilla Zea, N. Design of adaptive video games interfaces: A practical case of use in special education. Proceedings of 7th International Conference on Computer – Aided Design of User Interfaces (CADUI) (2008) 57-63.

Learner Modeling and Adapted Interaction in Educational Games

Kostas Maragos and Maria Grigoriadou

Department of Informatics and Telecommunications,
National and Kapodistrian University of Athens,
Panepistimiopolis, 15784, Athens, Greece
{kmaragos, gregor}@di.uoa.gr

Abstract. To enhance the effectiveness an educational game we have to adapt the system in order to provide one to one instruction and guidance according to individual educational needs. To achieve this we must build the learner model which is a model of the goals, preferences and knowledge of the individual student. In addition, the use of a pedagogical agent could lead to a more human and social learning environment and consequently in deeper learning. In this paper we present TALENT, a multiplayer educational game to teach elementary programming concepts. We describe the pedagogical elements of the game like the game world, the use of the learner model and the pedagogical agent. The combination of these elements supports the overall learning cycle and the adaptation techniques of the game.

1 Introduction

Games support the "learning by doing" pedagogical approach encouraging the growth of logic and the acquisition of skills and knowledge with a pleasant way [1]. Researches have already proved the advantages of integrating games in educational environments [2, 3]. Games constitute a source that motivates the learners to try and to develop their knowledge, while they put it into practice. In addition, they learn things that do not know at the same time they are engaged in an entertainment circumstance [4].

Incorporating educational computer games into an educational setting is not an easy task. We must take care of the curriculum, the educational objectives and the limited time of the lesson as well as the game level difficulty combined to the learners' ability. When the difficulty of the game is greater than the student's ability then the feeling of anxiety and disappointment arises [5]. On the other hand where the game difficulty is little then the student may feel bored [6].

Student's ability is a key factor for every educational environment. Today, researchers are aware that different students have different needs and learning styles. The learner preferences and learning styles, the previous knowledge and capabilities as well as student performance in the game, are factors used for the construction of his learner model. Based on that learner model the game may be able to adapt the content, presentation or even the guidance to the individual learner [7].

In addition, when the model is open to student which means that the model is a visible and interactive part of the learning environment this could give students power over their learning, a learning dialogue and motivation [8]. In a multiplayer game we can go one step forward allowing the student to compare the information presented in his model with that produced by the mean data of all the other students. This knowledge of the whole progress has been argued to motivate student to try to succeed more in the activities [9].

Multiplayer games are able to support social interaction that is argued to play a fundamental role in the process of cognitive development [10, 11]. Moreover, the utilization of a pedagogical agent a new opportunity to facilitate learning [12] could lead even to a more human and “social” learning environment [13]. Students interacting with pedagogical agents have been shown to demonstrate deeper learning and greater motivation [14]. The policy to combine the learner model with a pedagogical agent enhances the adaptation of the educational game providing also adaptive feedback to the learner.

In the next sections we’ll present TALENT (Teaching Algorithms EnvironmeNT) a multiplayer educational computer game for learning introductory programming concepts. Specifically, on section 2 we’ll describe the pedagogical elements and the core modules of the game and on section 3 we’ll discuss the interaction between these modules and the implemented method in order to provide an adaptive educational system centered on students educational needs. Finally, some conclusions and future work are discussed on section 4.

2 The Pedagogical Elements in TALENT

TALENT is designed as an adventure multiplayer online educational game. Adventure games offer powerful opportunities for learning and development of problem-solving abilities [15] and are suitable for science concepts that may be hard to visualize or manipulate with concrete materials [16]. In addition, adventure games which are consisted of tasks provide a clear progression overall in the game [17].

In this section we describe the core elements of the game designed according to pedagogical principles in order to improve learning, such as the multiplayer game world, the open learner model and the pedagogical agent.

2.1 The Multiplayer Game World

In TALENT students explore the world, communicate, exchange object with each other and are involved in programming activities in order to succeed in the game. An activity is the basic unit of the game structure and indicates a task to be performed. Several different activities are grouped together according to their learning goal forming activities groups. A student have to carry out successfully all the activities of

a group in order to advance to the next learning goal. This method of forming activities groups is the first step to support an adaptive educational game [18].
In order to support the above methodology we divided the TALENT game world into three tiers Map, Place and Mission. The Map tier is the main map of the game and is the main object that the student is facing when he enters the Game World. On the map there are several places that a student could follow. Places vary according their content. They might contain various learning units like presentations, examples and problems (missions) to help students achieve the learning goals. A mission in TALENT is actually a programming activity. According to the above methodology several missions provide a mission group that corresponds to a learning goal. An example with three places, seven missions and three different learning goals is shown in Fig 1.

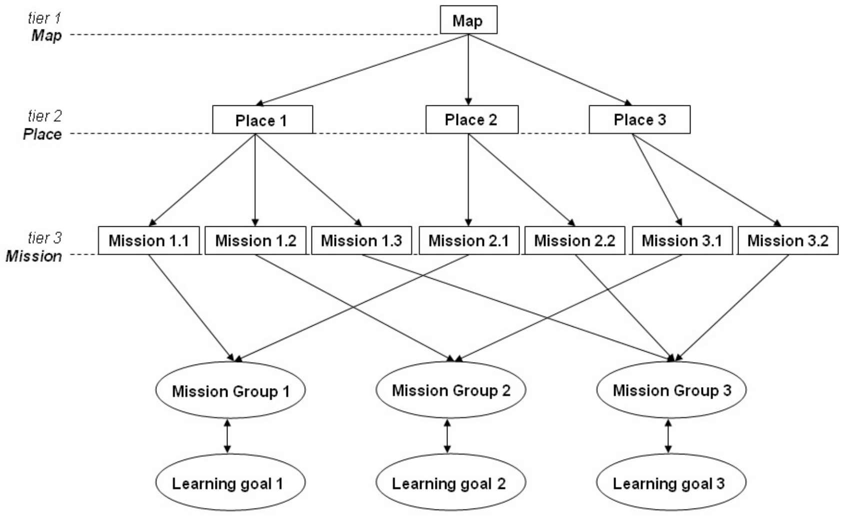


Fig. 1. Example presents three different learning goals. Each learning goal corresponds to a Mission group. Each Mission group is formed by different missions from three different places.

In a later section we'll describe in more detail how mission groups are used in TALENT and its adaptation mechanism.

Finally, the architecture of TALENT supports an open game world. This means that an expert or even a teacher will be able to alter the game world by using his own graphics, objects and learning units presented to students. To accomplish this critical feature all the Game World is based on database records and xml files and could be changed using the authoring tool of the environment.

2.2 The Open Learner Model

Adaptive educational systems build the learner's model in order to adapt the content and guidance of a system to the user's knowledge and goals [19]. Adaptation is used to decide whether to advance the learner to the next topic, to provide unsolicited advice and to find out learner's capabilities and the constructed knowledge so far [20].

The learner model engine in TALENT constructs the learner model based on information about learner's navigation, the use of tools, the progress in programming activities and the achievement of learning goals. Data that concerns the sequence of places and mission selection, the total time spent on a place or a mission, the total number of times that the student asked for help in a mission, the visits on a mission or a place, the number of times he communicated with the other students in the game and the progress on learning goals are saved in a separate database record associated with the student.

In addition, in TALENT the model is open to student. This means the learner model is by default viewable and changeable by students [21] unless prohibited by the administrator which could be the teacher of the class. Changing the learner model is a very important task for the correct functioning of the adaptation. To avoid arbitrary changes of a learner model the administrator has the ability to correct one or more learner models by using a special interface. Students are also able to compare their learner model with a group learner model [22] produced by the mean data of all the other students (Fig. 2).

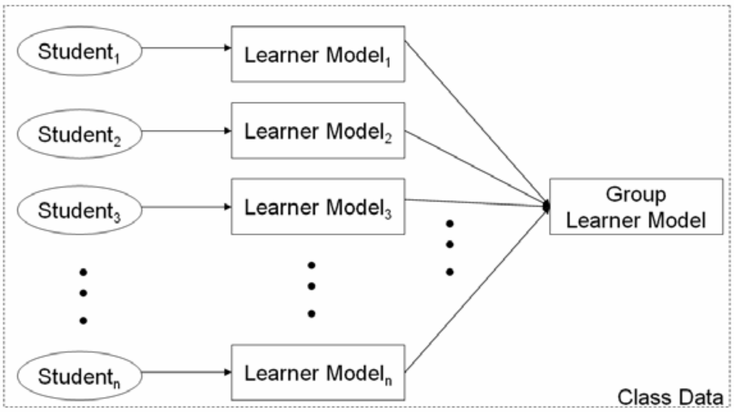


Fig. 2. The mean data of all Learner Models constructs the Group Learner Model

2.3 The Pedagogical Agent

A Pedagogical agent is defined as an agent that engages interaction with learners. Its use in educational systems is strongly suggested as it tends to increase the learner's motivation and engagement [23]. At the same time its pedagogical potential has already been proven [24].

The pedagogical agent that inhabits in TALENT is a mentor type agent providing not only information but also motivational support and guidance. Mentor type pedagogical agents leads to overall improved learning and motivation [25]. Furthermore, anthropomorphized agents play a social role [26] and put an emphasis on building social relationships with users. In addition students show differing perceptions according to agent characteristics [27]. For that reason when the student registers in TALENT he is free to choose between two different pedagogical agents, a man or a woman.

In the next section we describe the combination of the learner model and the pedagogical agent as well as the adaptation Technique in TALENT.

3 The Learning Cycle and Adaptation Techniques

Learning is a cyclic process and learning in an educational game is defined as a construction of cognitive structures through action in the game world [28]. As the student interacts with the game world in TALENT, the environment alters the learner's model. The pedagogical agent of the game relies on the information provided by the learner model to provide hints, motivation and guidance (Fig. 3).

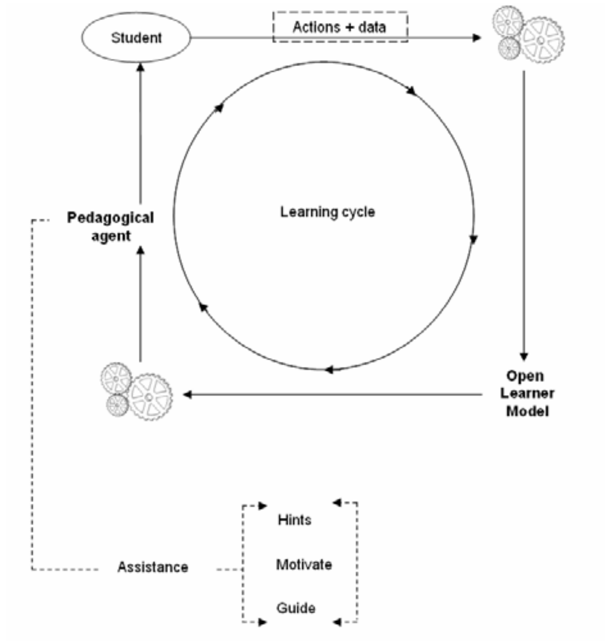


Fig. 3. Learning is a cycling process. Student data and actions build the learner model. The pedagogical agent consults learner model to provide assistance.

Adaptation takes place all over the game structure including content (presentations, examples and missions) and guidance.

For adaptive content presentation we use the explanation variants method [29] implemented by the page variants technique. With this technique, we keep two or more variants of the same page with different presentations of the same content. Each variant is prepared for one of possible user stereotypes (beginner, intermediate, expert). When presenting a page, the system selects the page variant according to the user stereotype.

For adaptive navigation support we use a combination of direct guidance and adaptive annotation techniques. In direct guidance the system decides what is the next best node for the user to visit according user's goal and other parameters represented in the user model [30]. In adaptive annotation the system augments the links with some form of comments which can tell the user more about the current state of the nodes behind the annotated links [31].

4 Conclusions

In this paper we have argued the potential of combining pedagogical characteristics like learner modeling and pedagogical agents with educational computer games. These two core characteristics will enable us to provide adaptation which tends to be a demand for the educational environments nowadays. In addition, the use of an open learner model will benefit the individual through the observation of the group learner model constructed by the mean data of all the students. The pedagogical agent that inhabits in the game watches the game world as well as the learner model in order to provide hints, motivation and guidance. Also, the system is able to adapt to students according to his learner model. The proposal for adapting to content presentation is based on page variants technique. On the other hand the adaptive navigation is based on a combination of two techniques, direct guidance and adaptive annotation.

Finally, we presented TALENT, a multiplayer educational game for teaching elementary programming concepts. TALENT utilizes learner modeling and pedagogical agent and provides adaptation according to the above specifications. A research with real data has already taken place. The analysis of the results will reveal the potential of the educational environment.

References

1. Klawe, M. & Philips, E. (1995), A classroom Study: Electronic Games Engage Children as Researchers*, Proceedings of CSCL '95 Conference*, Bloomington, Indiana, 209-213
2. Kirriemur, J., McFarlane, A. (2004), Literature review in games and learning., NESTA, Futurelab, 8
3. Mitchell, A., Savill-Smith, C. (2004), The Use of Computer and Videogames for Learning: A Review of the Literature. Learning and Skills Development Agency, Trowbridge, Wiltshire
4. Malone, T. W. (1980), What make things fun to learn? A study of intrinsically motivating computer games, Cognitive and Instructional Science Series, CIS-7, Xerox Palo Alto Research Center, Palo Alto
5. Sedighian, K. (1997), Challenge Driven Learning: A model for Children's Multimedia Mathematics Learning Environments, World Conference on Educational Multimedia and Hypermedia, Calgary, Canada
6. Csikszentmihalyi, M. (1975), Beyond boredom and anxiety: The experience of play in work and games. San Francisco
7. Cui, Y. & Bull, S. (2005). Context and Learner Modelling for the Mobile Foreign Language Learner, System 33(2), 353-367
8. Kerly, A. & Bull, S. (2007). Open Learner Models: Opinions of School Education Professionals, in K. Koedinger, R. Luckin & J. Greer (eds), Artificial Intelligence in Education, IOS Press, Amsterdam.
9. Bull, S. & McKay, M. (2004). An Open Learner Model for Children and Teachers: Inspecting Knowledge Level of Individuals and Peers. Intelligent Tutoring Systems: 7th International Conference, Springer-Verlag, Berlin Heidelberg, 646-655.
10. Vygotsky, L.S. (1978). Mind and society: The development of higher mental processes. Cambridge, MA: Harvard University Press.
11. Crawford, K. (1996) Vygotskian approaches to human development in the information era. Educational Studies in Mathematics. (31) 43-62.
12. Johnson, W. L., Rickel, J. W., & Lester, J. C. (2000). Animated pedagogical agents: Face-to-face interaction in interactive learning environments. International Journal of Artificial Intelligence in Education, 11, 47-78.
13. Hermans B. "Intelligent Software Agents on the Internet", FirstMonday, vol. 2, 3/1997.

14. Morozov M., Tanakov A., Bystrov D. (2001). Pedagogical Agents in the Multimedia Natural Sciences for Children, The 3rd IEEE International Conference on Advanced Learning Technologies, Athens, Greece, 9-11 July 2003 pp. 62-65
15. McFarlane, A. (ed.) 1997, Information Technology and Authentic Learning: Realising the potential of computers in the primary classroom. London: Routledge
16. Mitchell, A. & Savill-Smith, C. (2004) The use of computer and video games for learning: A review of the literature. Available: http://www.lsda.org.uk/files /PDF/1529.pdf (Accessed: 2005, July 25)
17. McFarlane, A., Sparrowhawk, A. & Heald Y. (2002) Report on the Educational Use of Games. Available: http://www.teem.org.uk/publications. (Accessed: 2005, July 18)
18. R. M. Carro, A. M. Breda, G. Castillo, and A. L. Bajuelos “A Methodology for Developing Adaptive Educational-Game Environments”, In P. De Bra, P. Brusilovsky, and R. Conejo(Eds.): AH2002, LNCS 2347, pp90-99. Springer- Verlag, Berlin Heidelberg 2002.
19. Brusilovsky, P. (1996), Methods and Techniques of Adaptive Hypermedia, User Modeling and User Adapted Interaction, v 6, n 2-3, pp 87-129
20. K. VanLehn.(1988). “Student Modeling”. In M.C. Polson and J.J Richardson, Foundations of Intelligent Tutoring Systems, (pp55-78).
21. Cook, R. and Kay, J. (1994), “The justified user model: a viewable, explained user model”, 4th International Conference on User Modeling, Hyannis, MA, pp. 145-150.
22. Kaplan, C., J. Fenwick, and J. Chen: 1993, ‘Adaptive hypertext navigation based on user goals and context’. User Models and User Adapted Interaction 3 (3), 193-220.
23. Mitrovic, A. and Suraweera P, (2000).. Evaluating an animated pedagogical agent. Proc. of ITS, Springer-Verlag. S. Antonio, TX, U.S.A.
24. Lester, J.C., B.A. Stone, and G.D. Stelling (1999), Lifelike Pedagogical agents for mixed initiative problem solving in constructivist learning environments. Journal of User Modeling and User-Adapted Interaction, 9(1-2).
25. Baylor, A. L. & Kim, Y. (2005). Simulating instructional roles through pedagogical agents. International Journal of Artificial Intelligence in Education, 15(1)
26. Baylor, A.L. and A. Plant, (2005), Pedagogical agents as social models for engineering: The influence of agent appearance on female choice.
27. Kim, Y. Pedagogical agents as learning companions: Building social relations with learners. In 12th International Conference on Artificial Intelligence in Education. 2005. Amsterdam, The Netherlands: IOS Press.
28. Kiili, K. (2005a). Digital game-based learning: Towards an experiential gaming model. The Internet and Higher Education, 8, 13–24.
29. Gonschorek, M. and C. Herzog: 1995, ‘Using hypertext for an adaptive helpsystem in an intelligent tutoring system’. AI-ED'95, 7th World Conference on Artificial Intelligence in Education, Washington, DC, pp. 274-281.
30. Brusilovsky, P. and L. Pesin: 1994, ‘ISIS-Tutor: An adaptive hypertext learning environment’. JCKBSE'94, Japanese-CIS Symposium on knowledge-based software engineering, Pereslavl- Zalesski, Russia, pp. 83-87.
31. Schwarz, E., P. Brusilovsky, and G. Weber: 1996, ‘World-wide intelligent textbooks’. EDMEDIA' 96 - World conference on educational multimedia and hypermedia, Boston, MA.

The Autopilot - A Personalized Pathfinder in Open Games for Learning

Franziska Spring-Keller[1] and Helmut Schauer[2]

[1] Educational Engineering Lab, University of Zurich, Switzerland, spring@ifi.uzh.ch,
[2] Educational Engineering Lab, University of Zurich, Switzerland schauer@ifi.uzh.ch

Abstract. Educators face many new challenges when it comes to the design of original games for learning. Especially so-called open games for learning are yet to be more investigated regarding the design for educational purposes. Open games for learning embed players within complex, open-systems and enable them to pursue their own goals. In such environments, players solve problems according to their individual backgrounds, preferences, and strategies.
This paper introduces a concept of dynamic feedback in so-called open games for learning. The autopilot takes over the role of an artificial player that can be activated by learners themselves in every situation they want. The idea is based on a former study that revealed learners that are so cautious in their behavior and their decisions that they either do not dare to try out things or do not experience essential events in the game that belong to intended learning goals.

1 Introduction

Although skepticism towards digital games as learning instrument is slowly fading away, it is still a young research field. Especially in the area of original games for learning, little is known of how to design these games and how to assess learning. Original means to design games specifically for educational purposes. To date, much attention has focused on role playing games, and educationally oriented games have focused on providing players an overarching "frame" of a profession or viewpoint, or learning through a professional identity (Shaffer, 2006). Less studied but frequently heralded, (Gee, 2003; Squire, 2008) are open or sandbox games that embed players within complex, open-systems and enable them to pursue their own goals. These kinds of games have been hypothesized to have strong educational potential. Examples of this category are *Sim City* (Starr, 1994), an urban planning game and *Civilization*, a game about world history (Squire, 2008). They are mainly suitable for learning about complex systemic behavior and cause-and-effect learning.
However, there are several reasons why open games for learning are so challenging to realize for education. Whereas most game designers for entertainment need to only ensure that a game is fun, educational designers also need to ensure that

educationally desirable learning is happening as well. Furthermore, different play and learning preferences produce new challenges for the design of open games for learning. The open environment of these games enhance personal learning paths but also endanger the risk of not achieving learning goals, desired by the educational designer.

A study, on which this paper is based on, about the analysis of player characteristics in open games for learning (Spring-Keller, in progress) confirmed these assumptions. In this study, an online game called *Hortus* was developed, specifically designed to investigate questions of how players would act and learn in open games for learning. *Hortus* is a strategy and simulation game about horticulture that teaches fundamental principles of biology. Learners take over the role of herbal healers who have to brew potions in order to heal sick people in a village. During their missions over five levels, they have to deal with obstacles, such as appearing bugs who destroy their harvest. Learners' actions were recorded and analyzed according to certain events and situations. User data was collected implicitly through the online game and explicitly through questionnaires and think aloud protocols. Player characteristics were derived from statistical analysis of quantitative information and qualitative evaluation of think aloud protocols.

Results revealed that certain players did not experience certain events in the game at all due to their cautious behavior. Others did not dare to experiment with certain features that would have allowed them to gain knowledge about specific characteristics of a plant. Daring to take risks and experimenting with game features are essential characteristics of learning within games (Gee, 2003). Furthermore, characteristics found in the game were not stable over time, only over certain situations or events.

These findings make it even more challenging to support learners in their personal needs and ways of solving problems.

In this paper, we introduce a concept of the autopilot that tries to address these issues. Cautious learners receive dynamic and personalized assistance when they require it. The autopilot provides situated assistance and is thus not bound to a distinct and stable learner profile.

As theoretical background, different models about adaptation are introduced and current research in this field is discussed. The functionality of the adaptive system with the autopilot is illustrated and the underlying theory for the learner model is explained. The concept of the autopilot will be demonstrated on the game *Hortus*.

2 Existing Approaches in Adapting Games for Learning

Adaptation in this context is interpreted in a wider sense and includes adjustments made by the system or activated by the learner. For instance, feedback or learning objects can be adapted to a learner's state.

There is a lot of literature about adaptation in digital learning environments and also games for entertainment. Research in adaptation and games for learning is still young.

The most common adaptive e-learning models are introduced in Burgos (2006) and Moedritscher et al. (2004). Those approaches are:

- *Macro Adaptation:*
  This kind of adaptation adjusts more general learning features according to a certain learner model. For instance, content presentation or learning goals are adjusted according to the learners' preferences and skill level.
- *Micro Adaptation:*
  This approach adjusts specific learning features on a micro level of the learning environment. This kind of adaptation is compared to one-on-one tutoring (Moedritscher et al., 2004). For micro adaptation, there are two steps of adjustment involved. First, the learners' behavior is analyzed and evaluated with quantitative methods. Second, instructions are initiated according to the results of this analysis. Response sensitivity plays an important role with this adjustment. This occurs through response times, click frequency, eye movements etc.
- *Aptitude-Treatment Interaction (ATI):*
  This approach adjusts instructions according to learner characteristics. The system proposes different types of instructions for different learner characteristics. They have generally three types of freedom in controlling learners. For low-prior knowledge learners, control is very limited and for learner with higher prior knowledge, there is much more freedom in controlling tasks and learning pace.
- Constructivist-Collaborative Approach:
  This model focuses on constructivist learning methods where the learner is actively involved in the learning process. This model is implemented in systems with virtual coaches and learning companions, respectively (Kort et. al., 2001).

Most successful systems use a combination of the above approaches. For instance, intelligent tutoring systems consider a mixture of Micro-Adaptation and Aptitude-Treatment Interaction (Moedritscher et al., 2004). Instructions are based on learners' characteristics but the adjustment happens from a moment-to-moment basis.
The autopilot is mainly based on micro adaptation but also includes features of macro adaptation.
In Burgos (2006), there are further adaptive approaches introduced that are more related to the content of adaptation. Eight different kinds of adaptation in e-learning are described that range from adjusting the interface to content-based adaptation over to adaptive feedback.
Most literature about adaptation in games for learning is concentrated on story-based games or on adaptive storytelling that should enhance learning. Commonalities of diverse literature in this field is the preservation of the flow experience and thus the flow of the storyline.

For this preservation, several architectural models have been developed that describe the interaction between the game and adaptive controller (Burgos et al., 2008; Law and Rust-Kickmeier, 2008; Peirce et al., 2008; Spring-Keller and

Ito, 2007). Most of the time, the game and the controller are separated. The practical reason is to keep the adaptive controller generic so that various instructional theories can be integrated without influencing the game structure. Further explanations about the most current architectural model can be found in Peirce et al. (2008).
Generally, controllers deal either with the adaptation of instructional feedback in the game (Law and Rust-Kickmeier, 2008; Peirce et al., 2008) or adaptation of scenarios according to a learner's skill level (Burgos, 2008).
However, there is a thin line of patronizing players with well intended instructional feedback and guiding learners towards the intended learning goal by supporting a learner's personal path. Games already provide situated and meaningful feedback as center characteristics (Gee, 2003). The challenge is to integrate educational valuable feedback so that the game play is not interrupted and to provide the illusion of freedom of choice.
As suggestion, the adaptive controller should only contain rules of how to adjust the game system itself according to learner characteristics. This is very difficult in story-based games. The story should not be interrupted and still needs to make sense despite adjustments in the game system. However, in open games, this problem is not as complicated as in story-driven games since there is usually only a loose story line. Therefore, open games are more convenient for this kind of adaptation. For instance, complexity could be increased or decreased. In *Sim City*, players have to handle multiple tasks in parallel. The regulation of traffic, for instance, is quite overwhelming for beginners. This feature could easily be introduced later in the game when players are more used to the environment of *Sim City*.

3 An Adaptive Game-Based Learning System

The goal of this adaptive game-based learning system is to improve the interactive communication between the system and learner. Thereby the game system takes the role of a coach or guide. Adaptation cannot occur by coincidence. It must be based on a specific learner model. Therefore, the game system is adapted dynamically in a given situation and based on the learner model. The learner model is derived from findings of the experiment with the game Hortus (Spring-Keller, in progress). In the study of Spring-Keller (in progress), results showed certain cautious behavior of learners that was repeatedly found in similar situations or events but could not be identified as stable trait. This means, adjustments have to be based on events or situations and cannot occur due to a general learner profile.

The adaptive system introduced here, focuses on assisting cautious learners and guiding them towards the intended learning goals. The functionality is shown in Figure 1.

1. The learner gives inputs through mouse clicks.

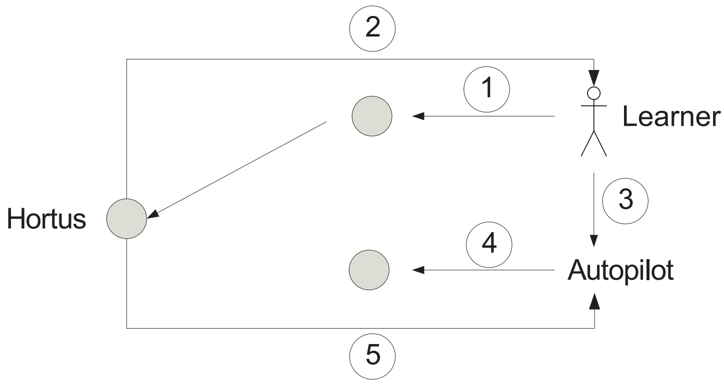


Fig. 1. Player characteristics are continuously analyzed and evaluated. The autopilot takes over playing the game if learners request it.

2. According to the learners choices, the game system provides visual feedback or by other multimedia means. Information of the game and the learner are stored in a database. Selected information such as click speed, objects clicked or certain game events are analyzed and categorized according to the learner model. This means, it also assesses learning progress of learners according to their knowledge.
3. Whenever learners need assistance in playing or making decisions, they can activate the autopilot. It takes over control of playing the game.
4. Every action of the autopilot is stored in the database as well. However, it is specially labeled as non-player actions.
   According to the respective characteristic, the information is sent to the control module which adjusts the respective game feature. For instance, if players did not experience a certain event, yet, a rule in the control module could be to adjust game parameters so that the player is experiencing this event.
5. According to the autopilot's choices, the game provides feedback. The communication cycle between autopilot and game system goes for a distinct time. After that, it either turns back to the starting point of the autopilot or it hands over directly to the player again. This functionality depends on the educational designer.

The advantage is that this adaptive game system is versatile enough to deal with continuously changing characteristics as well. Characteristics do not necessarily have to be stable over time. Cautiousness, for instance, is used for just specific situations. If a player changes his or her characteristics, the system adapts accordingly.
Learning progress is also not bound to a certain level in the game since open games do not provide a guided system. Thus, players reside in the game environment and explore different things without the need to repeat certain scenarios. Learning is embedded in events and actions that repeat from time to time.

4 The Autopilot - Dynamic and Personalized Feedback

If learners do not know how to continue playing the game or do not dare to try out something, they switch to the autopilot feature. This feature shows them a possible path through the game for a couple of steps like a movie. After the autopilot has stopped, the game either rewinds to the position/moment where it started or learners can continue where it stopped.

The idea is compared to a learning method called apprentice ship method (Steinkuehler, 2004) or cheat modes in games. Based on the idea of apprenticeship, the autopilot takes over the role of a trainer that teaches his or her students how to proceed in the game. As for cheat modes, for commercial games, there always have been cheat codes or cheat information around for specific situations. Most of these information were created by players themselves. For this concept, the cheat mode is integrated in the game as support system.

There are several games such as *Civilization* or *Sim City* that have an internal movie function. Players can watch their personal game in a replay mode. However, this feature usually is available at the end of the game and only from their own game.

The autopilot shows a possible path on-demand in the current situation before learners start playing. It can also be used as a "what if I did this" tool to figure out certain things without testing them and risking something.

The motivation behind this personalized pathfinder is to support people that got stuck in a certain situation and do not dare to experiment. The think aloud experiment in Spring (in progress) revealed certain types of people that are not used to experiment or do not want to experiment too much. If this attitude is age dependent or depends on gaming experience could not be evaluated to this point. Some learners needed more assistance in *Hortus* using the game help than others. There is the possibility that the group of non-gamers who aborted *Hortus* in a very early stage would have been better supported with this autopilot and thus would have dared to continue playing the game. However, the reason behind aborting the game is not proven and only a speculation.

Since the autopilot is started from every possible point in the game, "feedback" is highly situated.

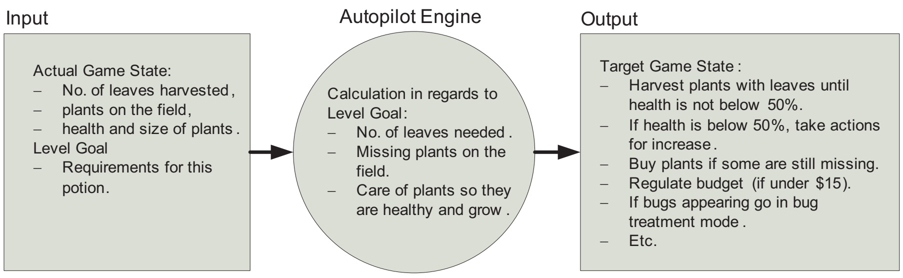


Fig. 2. Functionality of the autopilot on the example of Hortus.

The rule system for this feature only stores current information about the game and the learner. When the autopilot is started it retrieves the actual state of the game and the level goal and calculates the target state. It is calculated, for instance, how many leaves are left for brewing the potion. Depending on this information, a virtual player starts to take care of the plants in the field for a couple of turns until the autopilot stops. The goal as indicated in Figure 2 in the output window, serves as guideline where actions of the autopilot are targeted to.
There is a risk of overusing this feature so that learners only lean back and watch the entire game like a movie. This should be prevented with only a restricted use. One possible solution would be that every learner has a few credit points for using the autopilot. If these credit points are all used, they either can't use the autopilot anymore or they somehow have to earn credit points again. Earning credit points could be combined with a learning goal that has not yet been achieved. Hence, learners would not just hunt for points but learn something at the same time.
Learning goals are integrated into the autopilots' rule system. Hence, the movie stirs into the direction of teaching something. In *Hortus*, important learning goals were, for instance, to learn about plants characteristics and their special features when they grow big. Thus, if a plant already has a certain size, the autopilot could provoke a player attack to demonstrate this feature for the learner.

5 Conclusions and Future Research

There are many different approaches for adaptation in games for learning. This paper introduces a new concept, focusing on open games for learning, about an autopilot that provides situated and personalized strategic feedback to learners. It represents a kind of artificial player that takes over the game for a certain time.
The idea was formed as a result of a former study with the game *Hortus*. The results showed no stable traits for learner characteristics, such as cautiousness, in open games. They were very situated and only stable through certain situations or events in the game. Furthermore, Hortus enables different learners playing the game differently. This leads to uneven learning outcomes. The autopilot represents a guide to learners' that is seamlessly integrated into the game structure guiding them towards the intended learning goals.
As future research, there could be a combination of adapting the game environment according to the learner's characteristics, such as cautiousness and the autopilot. The autopilot does only suggest a possible pathway towards the goal, it cannot force learners to make the same choices. Changing the game environment would add another level of adaptation and thus could lead learner's more decisive towards the desired learning objectives.

References

Burgos, D. and Moreno-Ger, P. and Sierra, J.L. and Fernandez-Manjon, B. and Specht, M. and Koper, R.: Building Adaptive Game-Based Learning Resources: The Marriage of IMS Learning Design and ¡e-Adventure¿. Simulation Gaming. Vol. 39, no. 3, pp. 414-431 (2008)

Burgos, D. and Tattersall, C. and Koper, R.: Representing adaptive eLearning strategies in IMS Learning Design. In Proceedings of TENCompetence Conference, Sofia, Bulgaria (2006)

Gee, J.: What video games have to teach us about learning and literacy. Palgrave Macmillan, New York,(2003)

Kort, B. and Reilly, R. and Picard, R. W.: An Affective Model of Interplay between Emotions and Learning: Reengineering Educational Pedagogy – Building a Learning Companion. Second IEEE International Conference on Advanced Learning Technologies (ICALT'01). icalt, p. 0043, (2001)

Law, E. and Rust-Kickmeier, M.: 80days: Immersive digital educational games with adaptive storytelling. Proceedings of the First International Workshop on Story-Telling and Educational Games (STEG'08). The power of narration and imagination in technology enhanced learning. Netherlands,(2008)

Moedritscher, F. and Garcia-Barrios, V.M. and Guetl, C.: The Past, the Present and the Future of adaptive E-Learning – An Approach within the Scope of the Research Project AdeLE. In Proceedings of ICL, Austria, (2004)

Peirce, N., Conlan, O., and Wade, V. . Adaptive educational games: Providing non-invasive personalised learning experiences. Digital Games and Intelligent Toys Based Education, 2008 Second IEEE International Conference on, pp.28-35,(2008)

Shaffer, D.: How computer games help children learn. New York: Palgrave Macmillan,(2006).

Spring-Keller, F.: Player Characteristics in Open Games For Learning. Dissertation, University of Zurich, Switzerland, (in progress)

Squire, K.: . Open-ended video games: A model for developing learning for the interactive age. In K. Salen (Ed.) The Ecology of Games: Connecting Youth, Games, and Learning. The John D. and Catherine T. MacArthur Foundation series on digital media and learning. (167-198) Cambridge, MA: The MIT Press,(2008)

Spring-Keller, F. and Ito, T.: Combining personalisation and adaptation in game-based learning systems. In Proceedings of the Sixth Conference on IASTED international Conference Web-Based Education - Volume 2 (Chamonix, France, March 14 - 16, 2007). ACTA Press, Anaheim, CA, pp. 159-163, (2007)

Starr, P: Seductions of Sim: Policy as a Simulation Game. The American Prospect no. 17, pp. 19-29, (1994)

Steinkuehler, C. A.: Learning in massively multiplayer online games. In Y. B. Kafai, W. A. Sandoval, N. Enyedy, A. S. Nixon, and F. Herrera (Eds.), Proceedings of the Sixth International Conference of the Learning Sciences (pp. 521528), Mahwah, NJ: Erlbaum,(2004)

Authoring Environment for Story-based Digital Educational Games

Florian Mehm[1], Stefan Göbel[1], Sabrina Radke[2] and Ralf Steinmetz[1]

[1]Multimedia Communication Lab – KOM, TU Darmstadt
[2]httc Hessisches Telemedia Technologie Kompetenz-Center
Merckstr. 25, 64283 Darmstadt, Germany
{florian.mehm, stefan.goebel, ralf.steinmetz}@kom.tu-darmstadt.de, sabrina.radke@httc.de

Abstract. In this paper, the StoryTec authoring system, developed in the context of the European research project 80Days, is presented as an authoring tool enabling users without programming experience to create digital educational games (DEG) or stories and to integrate content into them. We provide an overview of related software as a motivation for the development of a comprehensive but easily learnable authoring tool for serious games which assists authors in different tasks encountered during the creation process. The architectural basis of the system is described, as well as a structured approach to story-based game authoring using the system, outlining each step of the process. The method and results of an early usability test carried out are presented.

Keywords: Interactive Digital Storytelling, Authoring Tool, Visual Programming, Serious Games, Technology-Enhanced Learning

1 Introduction

The authoring system StoryTec which is currently being developed in the context of the 80Days[1] research project, aims at providing diverse, non-specialist target user groups with a comprehensive yet approachable tool for creating serious games and interactive stories.
In an initial step, different potential user groups have been identified:

- Game designers (e.g. level designers, storyboard authors, concept developers) as the ‘creative people’ within the multi-step authoring and production process for story-based DEGs.
- Pedagogues and content providers as specialists in learning design, didactics and specific learning topics such as geography in the case of 80Days.

---

[1] 80Days – Around an Inspiring Virtual Learning World in Eighty Days. EU, FP7, IST, STREP, Challenge 4.1.2 Technology-enhanced Learning. www.eightydays.eu

- Teachers using the StoryTec framework to create small educational games as instruments in their courses.
- Technicians (e.g. game programmers) and content producers (e.g. modellers, graphic designers or audio specialists) who want to use StoryTec as a rapid prototyping environment for development and testing purposes.

Except for the last group, members of these groups are usually characterized by limited programming skills. Therefore, an easy and clear design approach (GUI layout and interaction design) as well as support in the form of templates should be provided. Especially members of the first and fourth group usually have some design background and do use design tools such as products in Adobe's Creative Suite in their daily work – or at least are aware of those tools and have a general idea how to use them. Subsequently, the design should consider typical underlying concepts, layout and GUI elements or interaction metaphors in order to provide a familiar look and feel.
In contrast to these groups, members of the second and third group usually do not have huge experience in those design tools, but are at least familiar with Microsoft Office products such as Word, Excel or PowerPoint.
As can be seen from this comparison, there are different expectations in the different potential user groups and it is necessary to find an appropriate compromise supporting all users as much as possible, which must include alternatives to scripting languages most game authoring tools use.
The focus of this paper lies on describing the authoring component of the overall StoryTec system. For a comprehensive overview and details on story execution see [6].

2 Related Work

The three major areas related to the StoryTec system are middleware software products for game development and e-learning as well as interactive storytelling tools. Furthermore, the system shares properties with visual/natural language approaches to programming. These related areas are presented in this section. A general overview of common practices in multimedia authoring is presented in [1].

E-Learning tools. In this category of tools, many research and commercial products are available which support authors in creating traditional e-learning courses. As an example, the Resource Center [8] features a web-based authoring toolset which can be used to create SCORM-compliant courses. A commercial example which also features the visual programming paradigm is Macromedia Authorware[2].
Game middleware. There exist a multitude of game creation tools with various degrees of complexity. One promising commercial example is the Unity3D[3] game creation software, which features a strong continuity between the authoring phase and

[2] http://www.adobe.com/products/authorware/
[3] http://unity3d.com/

the final game by offering a WYSIWYG view of the game during authoring. Another example which was introduced as a tool for teaching programming is the GameMaker system [17], introduced by Mark Overmars.

Storytelling tools. The focus of authoring tools for interactive storytelling lies on creating interactive experiences which are based on dramaturgic structures. Underlying methods and concepts of StoryTec are based on the theoretical and practical results achieved in the RTD projects art-E-fact (Cyranus authoring environment, see [9]), U-CREATE [7] and INSCAPE[4] [2].

Many current digital storytelling authoring systems make use of the emergent narrative approach, based on virtual agents. These include the Scenejo system [19], which is mainly focused on configuring conversational agents, approaches using rehearsals for demonstrating the envisioned story to virtual characters [14] or authoring during the playing experience [21]. Another project using the emergent narrative approach is the Façade system [15].

Visual/natural language programming. One factor limiting the approachability of the game development tools described above is their reliance on programming languages to customize the flow of events in a game when creating more than very basic games or when straying from simple included templates. Visual and natural language programming approaches can be utilized in this context. A very comprehensive survey of approaches is presented in [11].

Natural language programming allows users to write programs using languages similar to natural languages such as English. A broad overview of approaches is provided in [13]. One example relating specifically to game creation is the "Inform 7" [16] programming language for the creation of interactive fiction stories, in which story objects, scenes and interactions are described in a language resembling English.

Similar to natural language programming, the goal of visual programming languages is to create programs without the use of traditional programming languages. A system related to storytelling is the "Storytelling Alice" system as described in [10], in which a hybrid approach between natural language and visual programming is used to teach programming skills. The appropriateness of graph structures for authoring is supported in [20] in the context of authoring the behavior of virtual characters in the BECool system.

Examining the results of an analysis of related work in the fields of game and learning middleware, it becomes apparent that there exist few tools which integrate workflows for both game development and e-learning content production and integration. During the development of educational games, this results in a disconnection between pedagogues, who lack the skills for the actual game development, and game developers, who are not trained in pedagogy. Authoring tools which can connect the complex interactivity and graphical nature of serious games with support for pedagogical and narrative tasks are not researched in detail yet.

The design goals of the StoryTec system place it in the gap between these two, allowing better cooperation between both groups.

---

[4] INSCAPE – Interactive Storytelling for Creative People. FP6, IST, IP, www.inscapers.com

3 StoryTec Framework

This section describes the architecture of the StoryTec authoring tool, which acts as a framework for plugin suppliers. Figure 1 shows a broad overview of the components of the system.

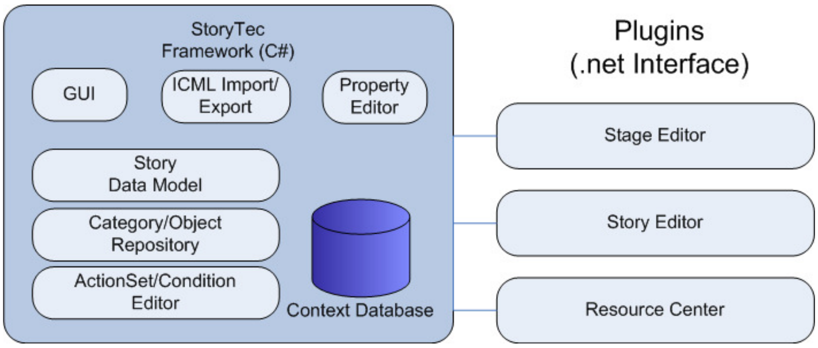


Fig. 1. Architecture of the StoryTec authoring system framework

As can be seen, the core of the system provides a set of central features which can be extended using plugins and which act as a programming framework for the development of plugins. Among the core components are the data model of the application or game the author is working on in the system, an export/import component for serializing the data model into the main export format of the system (ICML – INSCAPE Markup Language – as a story description language, see [6]) as well as a repository of objects available to the author (specific to the target platform, for example 2D or 3D assets, sound files or virtual characters). By providing this core functionality, plugin suppliers are assisted in creating conforming plugins and only need to extend core functionality when it is required (e.g. for providing exporters for platforms which do not utilise the Story Engine [6] which is targeted by the ICML language).

The framework must be augmented by plugins for the remaining core components, which are the Stage and Story Editor and the Resource Center. The Stage Editor component is similar to the WYSIWYG components of many of the authoring tools presented in section 2. Being immediately linked to one or more target output systems, this plugin defines the object categories the user can use in an application for this target platform (e.g. SWF files for a Flash-based player or 3D models for a 3D game engine). The Story Editor plugin is expected to visualize the structure of the story data model, while the Resource Center mainly visualizes the object repository as well as the Context Database, which provides both static and dynamic information such as game assets/props or event logs (see also section 4.5).

While externalizing the Stage Editor as a plugin is essential due to its close relationship to the target platform, the Story Editor and Resource Center were designed as plugins to provide the possibility of addressing new challenges (see the outlook below) while continuing to use older versions.

3.1 Graphical User Interface

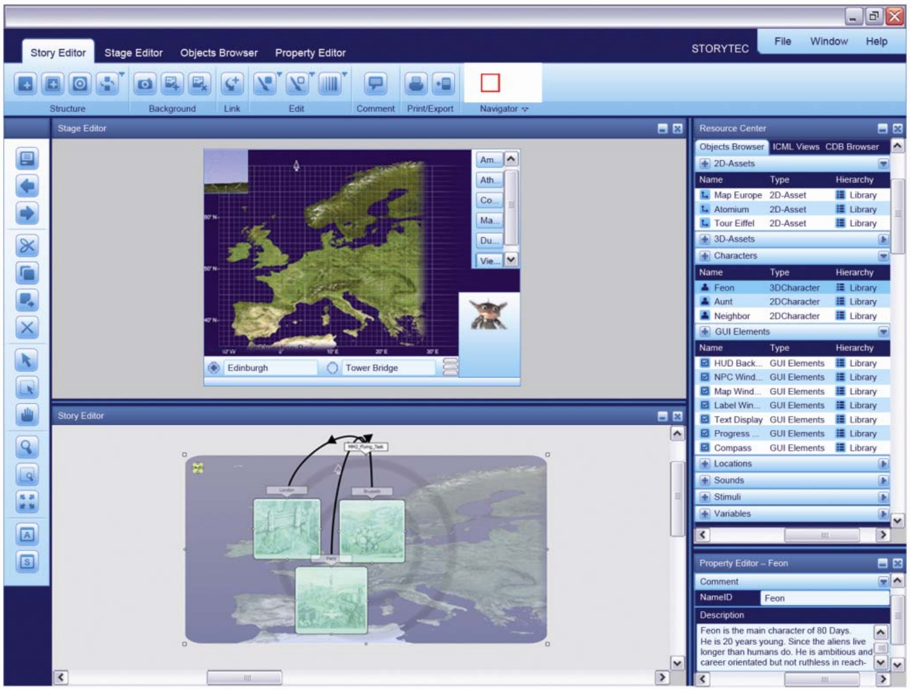


Fig. 2. The user interface of StoryTec (middle-upper part: Stage Editor, right: Resource Center and Property Editor, middle-lower part: Story Editor)

Utilising the WPF[5] framework, the main user interface of the authoring tool as well as styles available to plugins implement a user interface style adapted to the needs of the target user groups as identified and described in section 1. Therefore, an easy and clear design approach should be provided. Hence, the overall aim of the design was to reduce the complexity and presenting the multitude of functions the software provides in a simple and uncomplicated fashion.

As a result, the basic functions are united in the tool menu, which is reached via tabs in the top of the main window. The tool menu is context sensitive, adjusting its functions to the individual editors. This prevents the users from being inundated with information they do not need while working on a specific task. Functions shared by the different editors can be reached quickly via the toolbar on the left side at all times. Furthermore, limiting the colour scheme and making the icons more abstract helps in reaching the goal of reduction of visual complexity. To ensure that the users can adjust the user interface to their needs and the available screen space, flexible window functions are implemented.

[5] Windows Presentation Foundation, http://windowsclient.net/

4 Authoring Process

In this section, we present in general terms (i.e. without reference to a specific Stage Editor plugin) one possible structured process of creating a story-based educational game using StoryTec.

4.1 Creating a Story

After starting the StoryTec application, the user has the choice of loading an existing project, creating a new one from scratch or utilising one of the templates which are provided. A template provides the author with a pre-configured structure which is either based on a story model which can assist authors with little dramaturgic writing skills in creating a coherent and complete story (e.g. by conforming to the Hero's Journey story model) or on a project-specific structure which must be complied with (e.g. the structures a certain player component expects).

4.2 Constructing the Story Structure

If the users opted to create a project from scratch, a possible first step is defining the structure of this project. This can be achieved in the Story Editor component, in which the overall story is partitioned into individual scenes or complex scenes (see [6]). By drawing transitions as arrows between story units, the possible paths through the story are defined. Unconnected scenes should be interpreted as being freely combinable in an adaptive modular storytelling fashion, i.e. the most appropriate scene is selected during runtime, taking into account dramaturgic, learning and gaming aspects (see [5]). While other components like the Stage Editor are found in many of the systems described in section 2, the Story Editor as an abstract view on a story is a component that is an extension compared to many systems.

Scenes can be annotated in detail according to several categories. Authors can specify the expected time a user will remain in a given scene or which function of a story model a scene fulfils. A component visualizing which functions of the model are covered and which are missing in the story assists the user in creating a well-formed story. Skills which can be learned in the scene can be added to the annotation, along with skills the user should possess as prerequisites for the presented learning content.

4.3 Configuring Stages

Having defined the overall structure of a story, the user can continue by configuring the details of each scene. The first step towards this lies in defining which objects are placed in this scene. Similar to related tools as described in section 2, the Stage Editor features a view of the current scene in which the objects featured in the scene are visible in a WYSIWYG fashion. Users can drag objects from the Resource Center and drop them onto the Stage Editor or into the scene visualization in the Story Editor.

Objects will then appear and can be manipulated in both editors, with the Story Editor just displaying an abstract visualization.
Stage Editor plugins are not required to follow an exact WYSIWYG approach. For example, the Stage Editor developed for the 80Days project abstracts from the 3D gameplay by offering a 2D map on which geographical locations can be placed and interactions with them can be defined. This is an advancement from the INSCAPE Stage Editor which featured a very close correlation between the authoring phase and the final result.

4.4 Defining Actions in Scenes

After deciding which objects (such as virtual characters or props) will take part in a given scene, authors can define the logic that governs the flow of events in this scene. In comparable tools, this process is often realized either using a scripting language such as Python or a visual programming approach as in [10]. Since the design of StoryTec is primarily geared towards non-programmers, the latter variant was chosen.
In the ActionSet Editor (see Figure 3), the user can enter the sequence of events that will occur once this scene is encountered in the runtime environment by adding actions, represented by boxes, into a tree structure. Actions are always applied to objects which the author placed in the context of the current scene, therefore, only those objects are available in the ActionSet Editor. Furthermore, authors can trigger the transitions drawn in an earlier step, indicating that the active scene at runtime should change.

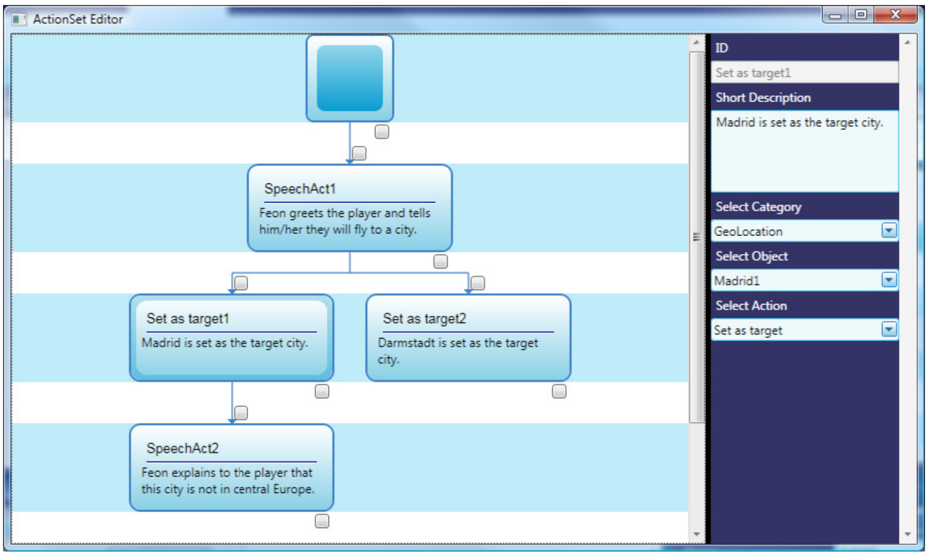


Fig. 3. The ActionSet Editor

In order to be able to react to user input and other events, Stimuli are added to the scenes as a special object type which can also have attached actions.

Branches are created by specifying the conditions for sub-trees. Using the Policy Editor (see [4] for the concept of Policies and Strategies in StoryTec), different game modes (e.g. beginner's/advanced mode or different stress levels) can be created, which can be taken into account when creating conditional branches.
Design challenges for the ActionSet Editor were the paradigms for interacting with the tree structure. For example, if a user deletes one action which is preceded and followed by a condition, the conditions would have to be concatenated in a sensible way. Furthermore, an important question is whether to restrict the ActionSet structure to be a true tree or to allow series-parallel digraphs, which are often used to visualize flowcharts and which are closer to programming languages, since they follow the convention that a program continues after a conditional block regardless of the condition of the block.

4.5 Testing, Iterative Design

After the first phase of authoring a story-based game has been completed in StoryTec, the game can be exported into the ICML format and be tested in a player application, for example in 80Days using the Nebula2 game engine as player. During testing, various information is saved into the Context Database, including the actual time users spent in a certain scene and the events which took place during a session. This information can then be imported back into the authoring system, where it is consolidated and can be used for example to update the expected time of scenes to be closer to an actual playthrough. This allows an iterative design process (compare with [3]) using which authors can improve their stories over several testing cycles.
The approach of iterative design of games or stories is further supported by offering text-to-speech-functionality to all plugins in the system, in order to test the effect of speech in the project before professional studio recordings are produced.

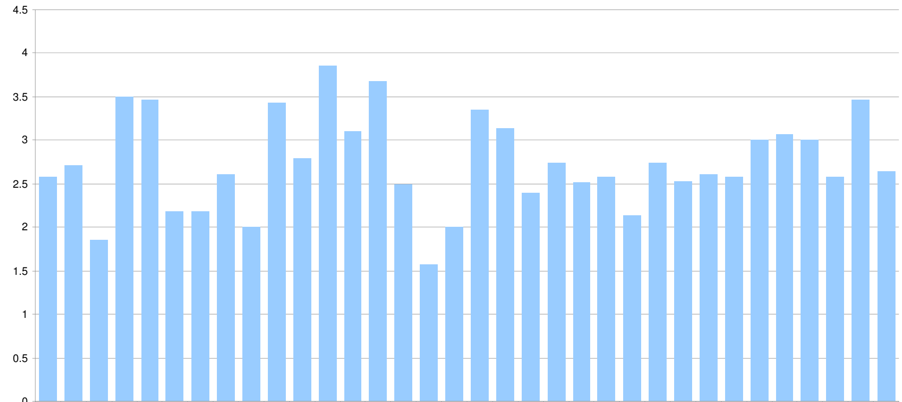
5 Evaluation

A first evaluation of the currently running StoryTec prototype was carried out with 29 students of which the majority were members of the Computer Science faculty. The participants tested the system in small groups of three. The test consisted of two phases, an usability test, which lasted around 40 minutes, and afterwards a questionnaire. Therefore, the test yielded both qualitative as well as quantitative information.
The first phase of the test was recorded by video and a protocol writer. By using an adapted version of the test method "Thinking Aloud" [18] the participants were asked to choose one person who would operate the system, with the other two instructing that person. By working through a task list, the participants built a small story in StoryTec, which could be played afterwards. By choosing this setup, the participants were encouraged to discuss possible ways of complying with the instructions and the monitoring staff could be made more aware of possible usability problems encountered by the subjects.

Overall, the interface, the functions and mainly the idea of StoryTec got a very positive rating. StoryTec was received very well by most participants, who described it as an innovative, user-friendly authoring tool. All of them were motivated to fulfil the task until the end. Nevertheless the test aided in discovering a range of small usability problems and technical bugs, which will be fixed in the next step. The result of the qualitative evaluation with the most impact was the lack of understanding the Stage Editor.

Furthermore, the aim of reduction of complexity of StoryTec by using a reduced design (e.g. fewer colours, abstract icons) could not be reached. On the contrary, the first test phase showed that this part of the design lead to problems in understanding the functions and the interface of the tool.

The quantitative test consisted of a questionnaire with 34 items which were presented with a six point scale from very bad to very good. The questionnaire included seven fields of software ergonomics (1. Suitability for the task, 2. Self-descriptiveness, 3. Controllability, 4. Conformity with user expectations, 5. Error tolerance, 6. Suitability for individualization and 7. Suitability for learning) by using the usability standard EN ISO 9241-10.

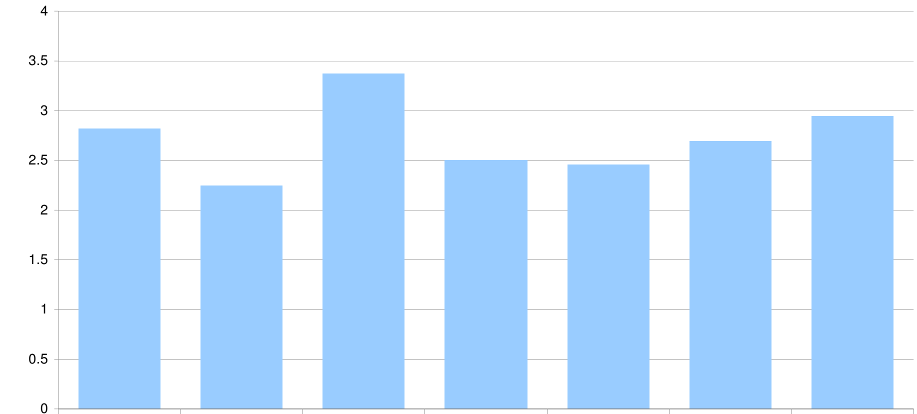


	1.1	1.2	1.3	1.4	1.5	2.1	2.2	2.3	2.4	3.1	3.2	3.3	3.4	3.5	4.1	4.2	4.3
Average	2.57	2.71	1.86	3.5	3.46	2.18	2.18	2.61	2	3.43	2.79	3.86	3.11	3.68	2.5	1.57	2
SD	1.32	1.46	0.97	1.04	1.32	1.44	1.44	1.37	1.12	1.4	1.29	1.24	1.17	1.12	1.45	1.43	1.27

	4.4	4.5	5.1	5.2	5.3	5.4	5.5	6.1	6.2	6.3	6.4	6.5	7.1	7.2	7.3	7.4	7.5
Average	3.36	3.14	2.39	2.75	2.52	2.57	2.14	2.75	2.54	2.61	2.57	3	3.07	3	2.57	3.46	2.64
SD	1.47	1.33	1.45	1	1.12	1.29	1.24	1.35	1.35	1.5	1.29	1.39	1.63	1.41	1.53	1.32	1.66

Fig. 4. Average results of the questionnaire

Figure 4 shows the average answers and the standard deviations of all items. Exemplarily, with an average of 3.86 (SD 1.24), the item 3.3 (addressing the controllability of the system) received the best rating. This implies that StoryTec quietly allows an easy change between menus and masks. The worst rating with an average of 1.57 was given to item 4.2 (addressing the conformity with user expectations of StoryTec). This implies that users were not sure if input they made was correctly executed or not by the system.



	Suitability for the task	Self-descriptiveness	Controllability	Conformity with user expectations	Error tolerance	Suitability for individualization	Suitability for learning
Average	2.821428571	2.241071429	3.371428571	2.5	2.457142857	2.692857143	2.95
SD	0.820730092	0.961003871	0.893953824	0.908397042	0.899147156	1.086594603	1.2479613

Fig. 5. The results of the questionnaire grouped by the 7 fields of software ergonomics as defined in EN ISO 9241-10.

Figure 5 shows the results of grouping the items into the seven fields of the usability standard EN ISO 9241-10. As can be seen, no direct conclusions can be drawn from the results concerning the software ergonomics of StoryTec.
While the evaluation of the questionnaires showed little evidence for certain strong or weak points in the design approach of StoryTec, the protocols of the first phase revealed many important details about the usability of the tool, which are currently being incorporated in an updated design and new paradigms.

6 Conclusion

The StoryTec authoring system is presented as a comprehensive authoring tool which addresses non-programmers and supports them in creating a story-based serious game with a coherent dramaturgic structure. The current version can be used to create projects which are run on the 80Days technical platform using the Nebula2 game engine as player.
Future versions will include several improvements. One area currently being worked on is the development of additional Stage Editors to support more target platforms. Among the possible platforms are web-based players using Adobe Flash, mobile appliances such as Nintendo DS, iPhone or Blackberry as well as 3D players using game engines or frameworks such as Unity3D or Microsoft XNA. The integration of more resources such as learning objects using SCORM is planned, too.
Furthermore, more research will be carried out in several areas. Among them are more advanced visual programming paradigms, especially in the ActionSet Editor, as well as supporting concurrency of Actions. In this context, the Condition Editor for configuring branches will be extended. Similarly, the default Story Editor will be

updated to better visualize important information during different stages of the authoring process, for example by visualizing the structure of the story in reference to the chosen story model or by hiding currently unimportant parts of the story. The visualization of skill structures (see [12]) underlying the learning aspects/goals of the story are planned as well.
In order to optimize the usability of the tool for the target audience, the results of the first usability test are currently being integrated, and new usability tests in later stages of development will be carried out.

Acknowledgements
The research and development introduced in this work is funded by the European Commission under the seventh framework programme in the ICT research priority, contract number 215918 (80Days, www.eightydays.eu).

References

1. Bulterman, D., Hardman, L.: Structured multimedia authoring. ACM Transactions on Multimedia Computing, Communications, and Applications 1(1), 89-109 (2005)
2. Dade-Robertson, M.: Visual Scenario Representation in the Context of a Tool for Interactive Storytelling. In: Cavazza, M., Donikian, S. (eds.) Virtual Storytelling 2007. LNCS, vol. 4871, pp. 3-12. Springer, Berlin / Heidelberg (2007)
3. Fullerton, T., Chen, J., Santiago, K., Nelson, E., Diamante, V., Meyers, A., Song, G., DeWeese, J.: That cloud game: dreaming (and doing) innovative game design. In : Proceedings of the 2006 ACM SIGGRAPH symposium on Videogames, pp. 51-59. ACM, New York, NY, USA (2006)
4. Göbel, S., Malkewitz, R., Becker, F.: Story Pacing in Interactive Storytelling. In Pan, Z., Aylett, R., Diener, H., Jin, X., Göbel, S., Li, L. (eds.) Technologies for E-Learning and Digital Entertainment. LNCS, vol. 3942, pp. 419-428. Springer, Berlin / Heidelberg (2006)
5. Göbel, S., Mehm, F., Radke, S.: 80Days: Adaptive Digital Storytelling for Digital Educational Games. Accepted to Second Workshop on Story-Telling and Educational Games, STEG (2009)
6. Göbel, S., Salvatore, L., Konrad, R., Mehm, F.: StoryTec: A Digital Storytelling Platform for the Authoring and Experiencing of Interactive and Non-linear Stories. In Spierling, U., Cavazza, M., Peinado, F., Aylett, R., Swartjes, I., Kudenko, D., Young, R., Tychsen, A., Pizzi, D., El-Nasr, M. (eds.) Interactive Storytelling 2008. LNCS, vol. 5334, pp. 325-328. Springer, Berlin / Heidelberg (2008)
7. Göbel, S., Salvatore, L., Konrad, R., Sauer, S., Osswald, K.: U-CREATE: Authoring tool for the creation of interactive storytelling based edutainment applications. In Cappellini, V., Hemsley, J. (eds.) EVA 2007 Florence Proceedings. Pitagora Editrice Bologna, Bologna (2007)
8. Hoermann, S., Hildebrandt, T., Rensing, C., Steinmetz, R.: ResourceCenter - A Digital Learning Object Repository with an Integrated Authoring Tool Set. In Kommers, P., Richards, G. (eds.) Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2005. pp. 3453-3460. AACE, Chesapeake, VA (2005)
9. Iurgel, I.: An authoring framework for interactive narrative with virtual characters. PHD Thesis, Technische Universität Darmstadt, Darmstadt (2008)

10. Kelleher, C., Pausch, R.: Lessons Learned from Designing a Programming System to Support Middle School Girls Creating Animated Stories. In : IEEE Symposium on Visual Languages and Human-Centric Computing, 2006, pp. 165-172. IEEE Computer Society, Washington, DC., USA (2006)
11. Kelleher, C., Pausch, R.: Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. ACM Computing Surveys 37(2), 83-137 (2005)
12. Kickmeier-Rust, M., Göbel, S., Albert, D.: 80Days: Melding Adaptive Educational Technology and Adaptive and Interactive Storytelling in Digital Educational Games. In Klamma, R., Sharda, N., Fernández-Manjón, B., Kosch, H., Spaniol, M. (eds.) Proceedings of the First International Workshop on Story-Telling and Educational Games. CEUR Workshop Proceedings (2008)
13. Knöll, R., Mezini, M.: Pegasus: first steps toward a naturalistic programming language. In: Dynamic Languages Symposium, Companion to the 21st ACM SIGPLAN symposium on Object-oriented programming systems, languages, and applications, pp. 542-559. ACM, New York, NY, USA (2006)
14. Kriegel, M., Aylett, R.: Emergent Narrative as a Novel Framework for Massively Collaborative Authoring. In Gratch, J., Prendinger, H., Ishizuka, M., Marsella, S., Pelachaud, C., Kopp, S., Kipp, M., André, E., Rehm, M., Aylett, R. (eds.) Intelligent Virtual Agents. LNCS, vol. 5208, pp. 73-80. Springer, Berlin / Heidelberg (2008)
15. Mateas, M., Stern, A.: Structuring content in the Facade interactive drama architecture. In Young, M., Laird, J. (eds.) Proceedings of the Conference on Artificial Intelligence and Interactive Digital Entertainment. AAAI, Menlo Park, CA. (2005)
16. Nelson, G.: Natural Language, Semantic Analysis and Interactive Fiction. Whitepaper (2005)
17. Overmars, M.: Teaching computer science through game design. Computer 37(4), 81-83 (2004)
18. Rosenbaum, S., Rohn, J., Humburg, J.: A toolkit for strategic usability: results from workshops, panels, and surveys. In: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 337-344. ACM, New York, NY, USA (2000)
19. Spierling, U., Weiß, S., Müller, W.: Towards Accessible Authoring Tools for Interactive Storytelling. In Spierling, U., Tychsen, A., Peinado, F., Stern, A., Riedl, M., Hitchens, M., Magerko, B., Manninen, T., Vallius, L., Kujanpää, T. (eds.) Technologies for Interactive Digital Storytelling and Entertainment. LNCS, vol. 4326, pp. 169-180. Springer, Berlin / Heidelberg (2006)
20. Szilas, N.: BEcool: Towards an Author Friendly Behaviour Engine. In Cavazza, M., Donikian, S. (eds.) International Conference on Virtual Storytelling. LNCS, vol. 4871, pp. 102-113. Springer, Berlin / Heidelberg (2007)
21. Thue, D., Bulitko, V., Spetch, M.: Making Stories Player-Specific: Delayed Authoring in Interactive Storytelling. In Spierling, U., Cavazza, M., Peinado, F., Aylett, R., Swartjes, I., Kudenko, D., Young, R., Tychsen, A., Pizzi, D., El-Nasr, M. (eds.) Interactive Storytelling. LNCS, vol. 5334, pp. 230-241. Springer, Berlin / Heidelberg (2008)

Gender Differences in Perceiving Digital Educational Games: A Mixed-Method Approach

Effie Lai-Chong Law[1], Tim Gamble[1] and Daniel Schwarz[2]

[1]Departmen of Computer Science, University of Leicester, LE1 7RH, UK
[2]TAKOMAT GmbH, Neptunplatz 6b, 50823 Köln, Germany
[1]{elaw|gamble}@mcs.le.ac.uk, [2]dan@takomat.com

Abstract. A survey on the initial design of a digital educational game entitled 80Days, which aimed to support the acquisition of geographical knowledge, was developed and administered to the target groups in Germany and England. Some significant gender and cultural differences in gameplay habit, game type preferences and game character perceptions were observed. Subsequent evaluative use-based studies on the executable prototype of the game showed that both genders could improve their geographical knowledge through the actual gameplay to a similar extent. However, the boys tended to have a higher level of positive user experience than the girls. In addition, qualitative data from two single-gender focus groups illustrated some interesting gender differences in perceiving various aspects of the game.

1. Introduction

Developing digital educational games (DEGs) that can cost-effectively foster learning with fun and pleasure is a *vision* for researchers and practitioners in the field of HCI and technology-enhanced learning. DEGs offer exciting and dynamic environments which engage players in meaningful and motivating learning activities, inspiring them to explore a variety of topics and tasks. Nonetheless, previous research suggests that children in general tend to find educational games uninteresting, and that gender difference existed with boys holding a more negative attitude towards edutainment games than girls (Kinzie & Joseph, 2008). While some recent research has indicated that gender-gap is getting closer (Hyde, 2005), whether such a gap-narrowing can be generalized to the domain of computer games remains unclear. In fact, the number of girls playing computer games has been increasing, but they still tend to be perceived as a masculine activity that more boys than girls prefer and spend time on. Such a disparity is attributable to the stereotypical presentation within games, a general lack of female characters in games, high competitiveness, and limited social interaction (Agosto, 2004; Gentry, Gable & Rizza, 2002; Hartmann & Klimmt, 2006). Even children in elementary schools perceive that software is gendered by design. The implication is more than just the attitude towards games; more serious impacts are girls' low confidence in working with computers and avoidance from technology-related fields (Gartner, 2006), adversely affecting their employability. Specifically, Kinzie and Joseph (2008) identified some interesting gender issues in game character preferences, for instance, the children in their study preferred characters to be of their same gender and ethnicity. Presumably, culture with its values, beliefs and norms plays an important role in shaping children's perceptions of game characters.

We are motivated to study gender differences in the context of a DEG under development. The prototype topic is based on geography. In the first phase, an initial game design concept was developed prior to any implementation. In brevity, the game story was about an alien kidnapping a boy and their flying round the world to collect relevant geographical information. A survey was designed to evaluate the acceptance of the target groups towards the game design, to verify if there are any gender and cultural differences in perceiving the game characters, and to elicit feedback on improving the game concept - a practical means to gather user requirements. In the second phase, an executable prototype was produced. User tests primarily in the form of observations and questionnaires were implemented to gauge the learning efficacy of the game, user acceptance towards it and different aspects of user experience. To further explore the issue of gender differences, two focus groups with representative school children were additionally conducted. Results from these three empirical studies (designated as Study 1, 2 and 3, respectively) relevant to gender issues are presented subsequently.

2. Related Work

Research questions addressing the influence of gender on attitudes towards computer games in general and on performance resulting from playing DEGs in particular are not new (e.g. Gorriz & Medina, 2002). However, answers to these questions keep on changing, given the highly dynamic landscape of gaming technologies. Besides, the computer game industry tactically lures more female to be frequent players. Broadly speaking, there exist two major types of factors – personal and technical - contributing to gender differences in computer gameplay patterns. On the personal level, traits (Bonanno & Kommers, 2008), motivation (Hartmann & Klimmt, 2006) and self-concept pertinent to IT competence (Hargittai & Shafer, 2006) are salient variables that interact intricately with game design features. Specifically, two genders are observed to differ in achievement needs with males generally demonstrating a higher level of desire to compete and beat their opponents than females (Hartmann & Klimmt, 2006), who seem disadvantaged and less effective in competitive settings such as computer games. A weaker competition orientation of females undermines their engagement in computer gameplay. Similarly, males are found to be keener sensation seekers than females as they tend to take risks (e.g. extreme sports) in pursuit of intense feelings and emotional arousal. The notion of sensation seeking has been widely adopted by Zuckerman (1979) and other scholars to explicate a range of social phenomena including various types of addictive behaviors. Arousal, a psychological construct underlying sensation seeking as well as gameplay, is normally at a higher level in males than females (Lucas & Sherry, 2004). These observations partly explain gender differences in gametype preferences and their different motivations to play. Males prefer games with confrontational and violent contents entailing fast responses and yearn to gain high scores, sense of control and other personal esteems. In contrast, females appreciate storylines and personalities of game characters to be explored at a relaxing pace and value building relationships with game characters or co-players (Agosto, 2004). Intertwining with competition and sensation seeking orientation is the issue of self concept. Despite insignificant gender difference in online abilities as indicated by some objective measures, females subjectively perceived such abilities to be much inferior to males (Hargittai & Shafer, 2006). Evidence on the trainability of cognitive-perceptual skills, which have traditionally been assumed to be innately stronger in males, seems not yet able to dispel the misconception in females.

In summing up the aforementioned arguments, presumably males tend to play games more engagingly than females; the former are then expected to show significantly higher learning gain from DEGs than the latter. However, recent empirical evidence indicates that no such gender difference can be detected (Papastergiou, 2009). Indeed, with the increasing awareness of gender differences and their underpinning factors, today's DEGs are so designed as to eliminate potential biases against any gender by incorporating a range of features and activities (Boyle & Conolly, 2008). Our project 80Days adopts a gender-sensitive approach by adapting the game to gender-based differences to optimize the learning process. Note, however, the elaboration of the adaptivity mechanism concerned falls outside the scope of this paper.

3. Method and Procedure

3.1 Study 1- A Survey on Initial Design Concepts

Design of the Questionnaire. The questionnaire consists of two major parts. Part A contains five close-end questions on the respondent's gender, age, gameplay habit, gametype preference, and affinity for geography. Specifically, four gametypes – learning, action, strategic and sport – are provided as options to reduce the possible confusion in children; the other taxonomies are deemed rather complex (e.g. Apperly, 2006). Part B addresses different aspects of the game. First a synopsis of the game story is presented. Then two close-end questions on the perceived interestingness of stories about aliens/UFO in general and of the game story in particular. An open-end question on describing improvement suggestions is presented. A set of four questions on understanding how respondents identify themselves with the story's main play characters are given. Another set of three questions on the preference of non-play character is posed. The last question is to assess the respondent's intention to play the game in the future.

Participants. Two samples from Germany and England were involved in the survey. They were school children aged between 11 and 14, the target group of the game. In Germany, the survey was conducted in the context of computer games fair. In England, the survey was administered in the classrooms of the five participating schools. Due to organizational constraints, the survey could only be conducted by the school teachers, who were asked to read aloud a script with similar wordings used in the German event. This step was taken to maximize the comparability of the data collected from the two settings.

Table 1. Demographic data of the survey respondents in the two countries

Country	Number/Age	Girls	Boys	Sub-total
German	Number	78	61	139
	Mean Age (SD)	12.6 (1.1)	12.8 (1.1)	12.7 (1.1)
British	Number	59	83	142
	Mean Age (SD)	12.5 (0.9)	12.7(0.9)	12.6 (0.9)
	Sub-total	137	144	281

3.2 Study 2 - User Tests of the Executable Prototype

Design of the User Test Session: It was conducted in groups of various sizes, ranging from 4 to 14, in the rooms within the respective school premises. Each participant was allocated to one computer where the game was installed and played it on an individual basis. One or two researchers were present in the rooms all the time to provide help and observe the participants' performance and behaviours. The arrangement of the test session is summarized in Table 2. The instruments listed therein have been developed by the project's research team.

***Participants*:** Two and four secondary schools in England and Austria were involved. Due to some technical problems, some of the participants could not complete the four missions. To compare validly the scores earned in Pre-test AoL and those in Post-test AoL, which were based on the contents of the four missions, our data analysis focused on the cases that successfully attempted all the missions. Besides, considering the differences in the test setting (e.g. larger group size in Austria) and curricular design, data of the British and Austrian samples are not merged whereas data from different schools in the same country are collapsed into one sample. In this paper, considering the length limit, we just report the findings on the British sample. Thirty-six children from the two British schools, of which the academic performances and infrastructure were comparable, could play through the four missions; the average age was 13 years old; 16 of girls and 20 are boys.

Table 2. Overview of the arrangement of a user test session

Activity	Objective and Instrument
Introduction	Describe the aim of the evaluation tests and instruct how to operate the laptops and headsets
Fill in the Background Questionnaire	Items: Identifier (ID), gender, age, gameplay frequencies, gametype preference, affinity for geography, subject grades, early involvement, and expectation
Fill in Pre-test Assessment of Learning (AoL)	16 domain-specific questions, open and close-ended, are based on the content of the game.
View Tutorial	6 open- and close-ended questions about the usefulness and usability of the tutorial material and presentation
Total Pre-Gameplay time: ~ 30 minutes	
Play each of the four micro-missions and fill in "After Mission Questionnaire" (AMQ) right away	Questions of AMQ are adapted to the content of the respective micro-mission. Research on user experience evaluation [X] suggests that data be collected as close to the interactive event as possible. Otherwise, the validity of the data may be compromised.
Total Gameplay time: ~52 minutes	
Fill in the Post-test Assessment of Learning (AoL)	The same questionnaire used for Pre-test. The rationale is to assess whether the children's knowledge of the geographical concepts covered in the game can be enhanced after playing it.
Usability and User Experience Evaluation of the Game Features	It consists of six sections with each of them focusing on different aspects of the game. The first section "General Game Experience" was adapted from [X].
Debriefing	Summarize the activities of the test session and thank the participants
Total Post-Gameplay time: ~33 minutes	

3.3 Study 3 - Focus Groups

Procedure: Prior to taking part in focus groups, participants were asked to play through the whole game without being required to fill in any questionnaire except the one for background data. Subsequently, focus groups were conducted as follows:

- Introduction: Participants were explained the purpose of the focus group
- Game recall exercise: Each participant was given a stack of Post-it notes and asked to write down whatever they could remember about the game.
- Sharing game recollections: Participants were presented three A3 sized sheets, one for each: "Positive" (green), "Neutral" (yellow) and "Negative" (red). They were asked to stick their notes to the respective sheets based on their own judgment how to categorise their notes.
- Guided discussions on different aspects:
  - Gameplay, e.g. "*In the whole game, which game character do you think you are supposed to be and which one would you like to be?*" (NB: the rationale is to understand if there is any mental gap in role adoption)

- Game characters and game story, e.g. "*How would you change the alien so that you will like him better*?"
- Learning part, e.g. *"How would you compare learning geography through the game with through normal classroom teaching?"*
- Geographical content, e.g. "*If you could add any aspect of Geography to the 80Days game, what would it be and how would you do it*?"

- Debriefing

Participants: Two single-gender groups, five boys and five girls, from a British secondary school (different from that in Study 2) were involved. Their participations were voluntary. The average age was 13.4 for the female group and 14.0 for the male one. All the participants except one girl, who had never played computer games before, were frequent gamers.

4. Results and Discussion

4.1 Study 1 - Survey

Results show that half of the British boys (52%) play games everyday and half of the German boys (51%) play games more than twice per week. Interestingly, 14% and 12% of the British and German girls report that they have never played games, whereas all of the British boys have played games. 45% of the German girls play games less than once per week whilst 44% of their British girls play more than twice per week. These figures seem to suggest that (i) Boys tend to play games more frequently than girls, irrespective of the country of residence; (ii) the British children tend to play games more frequently than their German counterparts. To investigate whether these observations are statistically significant, we performed the linear categorical regression analysis. The value of $R^2 = .25$ indicates that the two predictor variables *gender* and *country* can explain only 25% of the variations of the gameplay frequencies. Results show the significant effect of the predictor *gender* (beta = .49, $t = 9.32$, $p<.001$)) and the non-significant effect of the covariate *country* (beta = -.017, $t = -.136$, $p>0.05$). Boys tend to play games more frequently than girls, and the country of residence does not have a strong effect on the children's gameplay frequency.

Cramer's V was used to evaluate if *gender* was associated with *gametype preferen*ces. The most preferable gametype for both the British girls (51.7%) and boys (49.5%) are Action, followed by Strategic and Sport. The least preferable gametype is Learning with only 3.2% and 2.2% for the girls and boys, respectively. The value of the Pearson chi-square equals 0.581 (p =. 901), indicating that *gender* and *gametype preference* for the British sample are **not** significantly related. In contrast, the German sample demonstrates a slightly different pattern from their British counterparts. The most preferable gametype for the German girls is Strategic (40.7%), followed by Action and then Sport; the most preferable gametype for the German boys is Action (54.3%), followed by Strategic and then Sport. The least preferable gametype is Learning with 13.2% and 3.2% for the girls and boys, respectively. The value of the

Pearson chi-square equals 13.972 (p = .003), indicating that *gender* and *gametype preference* for the German sample are significantly related.

With the aim of evaluating to what extent the respondents tended to associate the Boy's (the main play character) attributes with their own, they were asked to rate first the Boy and then themselves, using a 7-point scale, with respect to six pairs of contrasting adjectives adapted from the instrument Speech Evaluation Instrument [4] consisting of three subscales – superiority, attractiveness and dynamism, against which the entity of interest is evaluated:

- *Superiority*: Intelligent vs. Unintelligent; Uneducated vs. Educated;
- *Attractiveness*: Friendly vs. Unfriendly; Cold vs. Warm;
- *Dynamism*: Peaceable vs. Aggressive; Talkative vs. Shy

The exercises resulted in a set of so-called "Boy-based ratings" and another set of "Me-based ratings". We computed the correlations among them independently for the German and British samples. A number of statistically significant correlations are found. Nonetheless, based on our research interest, we explore to see whether there are gender differences in perceiving the relationships between the Boy's attributes, between the Me attributes, and between these two sets. Interestingly, results consistently show that the German female respondents tended to perceive the attribute interrelations, be they applied to the Boy or themselves, in a more complicated manner than did their male counterparts. Presumably, the German male respondents may associate their own attributes with the Boy's (same gender) more strongly than the female respondents (opposite gender) do; the empirical results indicate otherwise. In contrast, the British respondents' perceptions, irrespective of gender, are less complicated than those of their German counterparts. Interestingly, the British male respondents tend to perceive the associations in a more complex way than their female ones – a reversed trend demonstrated by the German sample. Fig. 1 and Fig. 2 illustrate the results how the respondents perceive the associations between the game main play character ("Boy") and themselves ("Me"). Contrasts are observed across gender and culture. We also aim to find out whether those who perceived a stronger "Boy-Me" association might have a higher tendency to play the game in the future (i.e. the last question of the survey) by summing the absolute differences in ratings over the six pairs of adjectives. While there is a moderately significant correlation for the British sample ($r = -.24$, $N = 199$, $p < .05$), it is not significant for the German sample.

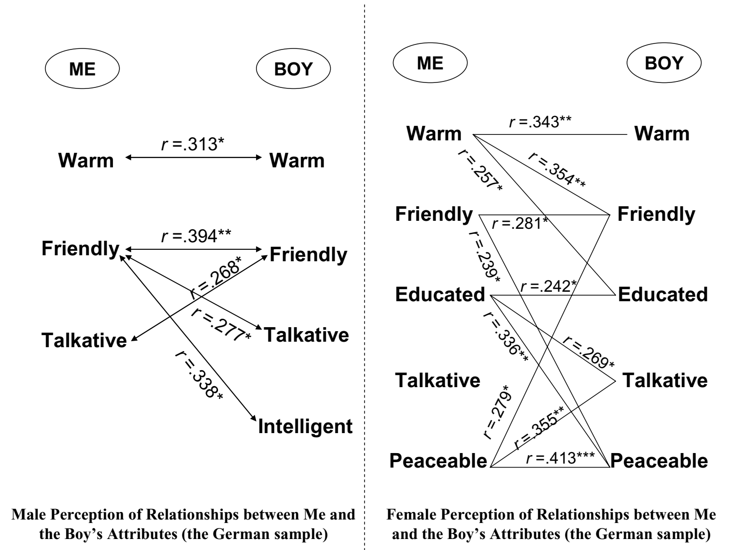


Fig. 1. Gender-specific perceptions of the play character and oneself (German sample)

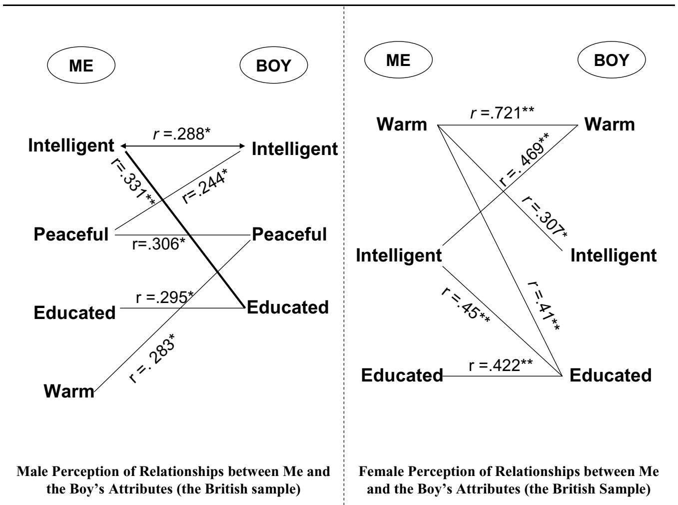


Fig. 2. Gender-specific perceptions of the play character and oneself (British sample)

4.2 Study 2: User Tests of the Executable Game Prototype

Our basic assumption is that by completing the four Missions of the game the participating children can gain better understanding of the geographical contents addressed therein. The improvement can be measured in terms of the significant difference in their performance between the Pre- and Post-Assessment of Learning Questionnaire (Pre-ALQ vs. Post-ALQ). The British participants demonstrated statistically significant learning gains (Pre-ALQ: mean = 20.8; Post-ALQ: 27.4; t =

5.25, *df* = 35, *p*<.001). When breaking down the data by gender, some interesting observations are obtained. In both Pre-ALQ and Post-ALQ, the boys performed significantly better than the girls (*p* <.05). The girls gained on average 5.1 points with the range of difference being -6 to 14 whereas the boys gained on average 7.8 points with the range of difference being -9 to 23 (Note that some children lost rather than gained points after the gameplay; we speculate that either they made guesses in Pre-ALQ or got confused about certain concepts during the gameplay). But the boys did *not* improve to a significantly larger extent than did the girls. In other words, both genders benefit from the gameplay in terms of knowledge gain, but it did not privilege the boys or frequent gamers (i.e. gameplay frequency is a non-significant covariate).

Existing literature suggests that evaluation of children game experience should address seven dimensions, including *challenge, competence, flow, immersion, negative affect, positive affect and tension* (Poels et al., in press). Accordingly, 14 statements are adapted for evaluating our game, two for each dimension (Table 3). The participants are asked to rate each of them with a 5-point Likert scale with the rightmost and leftmost anchors being ‘not true at all’ and ‘very true’, respectively.

Table 3. Seven dimensions of general game experience

S1.	Playing this game was useful for me to learn geography	(Challenge)
S2.	This game was interesting for me	(Positive Affect)
S3.	I put a lot of effort in playing the game	(Challenge)
S4.	Playing this game was a waste of my time	(Negative Affect)
S5.	I felt frustrated when playing the game	(Negative Affect)
S6.	I felt proud when I finished the game	(Competence)
S7.	The game was too difficult for me	(Competence)
S8.	I could concentrate easily on the game activities	(Flow)
S9.	I had the feeling of controlling the game	(Positive Affect)
S10.	I was completely absorbed by the game	(Immersion)
S11.	I felt exhausted after playing the game	(Tension)
S12.	I had the feeling that I had returned from a journey	(Immersion)
S13.	I felt time pressure	(Tension)
S14.	I was fast at reaching the target of the goal	(Flow)

Results indicate that significant gender differences can only be found in two dimensions, namely *Competence* (S7) and *Flow* (S8, S14). The girls rated themselves lower in Competence than did the boys (M*g* = 2.3, M*b*=1.4; *t* = 2.4, p<.05; NB: the higher the rating in S7, the lower the perceived competence is). In contrast, the boys rated significantly higher than did the girls in S8 (M*g* = 3.1, M*b*=4.0, *t* = 2.4, *p*<.05) and S14 (M*g* = 3.1, Mb= 4.0, *t* = 2.5, *p*<.05). As the feeling of flow [x] is imperative for engaging in gameplay, it can be inferred that the boys had stronger positive experience through playing the game than did the girls.

4.3 Study 3 – Focus Groups

The focus groups were audio-taped and transcribed in verbatim. Some interesting gender-specific findings are obtained. In game recollections, only two boys named a negative feature: the character Aunt (i.e. a static 3D female figure presenting geographical information via a text window) and controls of the Spaceship. In contrast, all the five girls named at least two negative features, including the Aunt and the Spaceship. This observed differences suggest that the boys had a higher acceptance towards the game than did the girls. In guided discussions, all the five boys and only one girl considered flying UFO the most positive aspect. Three girls appreciated the graphics the most and criticized that there was too much talking in the game. With regard to the role adoption, it seemed that both genders had difficulty to recognize that they were supposed to play the role of the abducted Boy. One girl, who was able to do so, uttered: “You were the boy. You were looking through his eyes” whereas another stated surprisingly: “I never even know we were playing with him. I didn’t even know we were that person.”

Concerning the learning part, both groups mentioned the importance of getting an explanation. Some girls remarked: “The teacher gives you more of an explanation”, “… you can ask them (the teacher) if you’re stuck. Similarly, a boy mentioned “In the classroom you’ve got an explanation… but in the game it just tells you you’ve got to go here and name the countries, like it doesn’t give you an explanation.” Interestingly, some boys suggested lengthening the missions but some girls suggested shortening them. Quite unexpected none of the girls proposed including a female game character (cf. the Boy) whereas one boy recommended providing a choice of a male or female play character.

5. Concluding Remarks

Previous research suggests that children, especially boys, tend to find learning games boring. It is corroborated by our findings of Study 1 that among the four gametypes learning game is least preferable and girls are more positive towards it than boys. Existing research also suggest that children tend to prefer game characters that are in some way "like me". Cultural preferences for normative personal qualities may influence children's preferences for the characters they play. While there are some very interesting gender and cultural differences in interpreting the main play character's qualities and in associating those qualities to theirs, such associations do not affect their intention to play the game. The setting where the survey was conducted could have impact on the children's perception and acceptance of the game: the relaxing atmosphere in the game fair with the exhibitors as opposed to the more structured classroom environment with the teacher. Results of Study 2 suggest that both genders could benefit to a similar extent from playing the game in terms of domain-specific knowledge gain. Interestingly, the female participants found the game difficult to play than did the boys. In other words, the girls' perceived competence in gameplay was significantly lower than the boys'. This observation is consistent with the earlier research. In the same vein, the boys had experienced a significantly higher level of flow feelings, which are important for engaging in and enjoying a game. Findings of the two single-sex focus groups also suggest gender-specific likes and dislikes towards different aspects of the game. Surprisingly, the girls tended to be more critical. Currently, we explore psychosocial theories to explicate the phenomena observed and their implications on future work.

References

- Agosto, D. (2004). Girls and gaming: A summary of the research with implications for practice. *Teacher Librarian, 31*, 8-14.
- Apperley, T.H. (2006). Genre and game studies: Toward a critical approach to video game geners. *Simluation & Gaming, 37*(1), 6-23.
- Bonanno, P., & Kommers, P.A.M. (2008). Exploring the influence of ender and gaming competence on attitudes towards instructional games. *British Journal of Educational Technology, 39* (1), 97-100.
- Boyle, E., & Conolly, T. (2008). Games for learning: Does gender make a difference? In T. Conolly & M. Stansfield (Eds.), *Proceedings of the 2nd European Conference on Games Based Learning* (pp.69-75). Reading, UK: Academic Publishing.
- Gartner Inc. Report. (2006). *Women and men in IT: Breaking through sexual stereotypes*, 5-9 November 2006, Gartner ITexpo Symposium, Cannes, France. Online at: http://www.gartner.com/it/sym/2006_/esc18/esc18_home.jsp
- Gentry, M., Gable, R. K., & Rizza, M. G. (2002). Students' perceptions of classroom activities: Are there grade-level and gender differences? *Journal of Educational Psychology, 94*(3), 539-544.
- Gorriz, C. M., & Medina, M. (2002). Engaging girls with computers through software games. *Communications of the ACM, 43*, 42-49.
- Hartmann, T., & Klimmt, C. (2006). Gender and computer games: Exploring females' dislikes. *Journal of Computer Mediated Communication, 11,* 910-931.

- Hargittai, E., & Shafer, S., (2006). Differences in actual and perceived online skills: The role of gender. *Social Science Quarterly, 87*, 432-448.
- Hyde, J.S. (2005). The gender similarities hypothesis. *American Psychologist, 60*(6).
- Kinzie, M.B., & Joseph, D.R.D. (2008). Gender differences in game activity preferences of middle school children. *Educational Technology Research & Development, 56,* 643-663
- Lucas, K., Sherry, J. L. (2004). Sex differences in video game play: A communication-based explanation. *Communication Research, 31*, 499-523.
- Parastergious ,M.(2009) Digital Game-Based Learning in high school Computer Science education: Impact on educational effectiveness and student motivation. *Computers & Education, 52*(1). 1-12.
- Poels, K., IJsselsteijn, W.A., de Kort, Y.A.W., and Van Iersel, B. (in press). Digital Games, the Aftermath. Qualitative insights into Post Game Experiences. In R. Bernhaupt (Ed.), *Evaluating User Experiences in Games*. Berlin: Springer.
- Slater, M.D. (2003). Alienation, aggression and sensation seeking as predictor of adolescent use of violent film, computer, and website content. *Journal of Communication, 53*, 105-201.
- Zahn, C.J., & Hopper, R. (1985). Measuring language attitudes: The speech evaluation instrument. *Journal of Language and Social Psychology, 4*, 113-122.
- Zuckerman, M. (1979). *Sensation seeking: Beyond the optimal level of arousal*. Hillsdale, NJ: Erlbaum.