A narrative metaphor to facilitate educational game authoring

Eugenio J. Marchiori [1], Javier Torrente [1], Ángel del Blanco [1], Pablo Moreno-Ger [1], Pilar Sancho[1], Baltasar Fernández-Manjón [1,2]

[1] *Department of Software Engineering and Artificial Intelligence, Complutense University, Madrid*

[2] *Laboratory of Computer Science, Massachusetts General Hospital, Harvard University, Boston*

(emarchiori, jtorrente, angel.dba, pablom)@fdi.ucm.es, pilar@sip.ucm.es, balta@fdi.ucm.es

Abstract

In this paper we present WEEV (Writing Environment for Educational Video games), a methodology for educational *point-and-click* adventure game authoring. Our approach aims to allow educators to actively collaborate in the educational game development process, using a narrative-based representation. WEEV is based on a pragmatic reinterpretation of previous works on narrativity and video games, enhanced by the use of a novel visual language to represent the flow of the story or narrative. The WEEV methodology has been implemented into an actual tool based on the already established <e-Adventure> platform for educational games. This tool was improved with feedback gathered from formative evaluation, end-users testing (i.e. educators), and actual use in the development of an educational game. The system, still under development, presents some user-interaction problems along with a need for the educational effectiveness of the resulting games to be further analyzed. However, this paper highlights that, according to the qualitative results of evaluations, WEEV can indeed be successfully applied to simplify the game creation process and that by using representations of games that educators can understand, WEEV can help provide educational value to games.

Keywords

authoring tools and methods; serious games; improving classroom teaching

1. Introduction

Educational video games, or serious games, are continually growing in number and complexity, as shown by growth in the interest and investment in them (Wexler, Corti, Derryberry, Quinn, & Barneveld, 2008). These video games imply substantial changes in the way that educators teach and how they use the teaching material. A serious game is usually a "black box" that educators must use "as is" in their courses. However, as educators are used to having more control over other course materials (e.g. creating, modifying or adapting course contents) the inclusion of a non-customizable teaching tool can be a drawback.

Game-authoring tools can provide a way for educators to recover their place in the creation or adaptation of contents when video games are used in a classroom. Through these tools educators can take an active role in development, by either creating the games, or contributing modifications or suggestions to expert teams in charge of the actual development. In this manner, educators can ensure that games provide learning opportunities and usefulness (understood from an educational perspective), aspects that

have the greatest impact on students' preference for games (Bourgonjon, Valcke, Soetaert, & Schellens, 2010).

Game authoring tools can also help educators feel more comfortable while using educational games, by allowing them to understand the inner workings of the game and make small modifications as they see fit. This allows educators to master the content and make it theirs, which is quite commonly a critical factor in the adoption of third-party content by educators.

Traditional video game development tools are usually based on programming concepts unfamiliar to many educators. These tools usually inherit not just the concepts and wording used in programming languages (e.g. *3D models, skeleton animations, Boolean conditions*), but also the complexity, in terms of user interaction, of interactive content authoring tools designed for experts. For example, many tools classify elements by their interactivity, understood as how elements are used (e.g. what game objects do for different keyboard inputs), and not by their function in the game (i.e. their role in the story).

In tools created for programmers, defining how game elements are used takes precedence over defining what they are used for. . However, this is not the most adequate structure for education professionals whose goal it is to teach through the game rather than to program it. As the function of an element usually depends on several interactions (e.g. show the question, answer the question, increase the game score, start the next part of the story, etc.), it usually becomes hard to understand by non-experts. Nonetheless, a correlation between different interaction mechanisms to certain functions in the game can be established (e.g. a typical in-game multiple-choice question in a conversation is often used to evaluate the knowledge of a given concept). In many cases, the interaction with the game can sometimes be automatically inferred from the function its elements must play. This suggests that an increased focus on function does not necessarily imply a limitation on interactivity, but it does imply a change in point of view during development.

In addition, most authoring tools use content-based or object-centered descriptions that result in the logic of the game being scattered around different components in a story that emerges from the interaction further complicating the problem. This means that the story is implicit, cannot be viewed as a whole, and is difficult to understand. For instance, the consequence of player action might be the modification of a variable, which only later, and indirectly, triggers the modification of an art resource that results in the player seeing the consequence of the action in the game world.

We propose bringing game development closer to the story-writing process by using a description based on a narrative metaphor (i.e. development based on the function of the game, more natural to non-developers such as educators). Some authors such as Ryan (2006) consider video games to be a new narrative medium. Educators, though usually unfamiliar with programming, can be expected to be used to other *narrative* materials (e.g. stories). This narrative metaphor is inspired by different heuristics, case studies and proposals regarding story-writing and video game development. We use an explicit representation of the story to create a comprehensive description of the game, "weaving" the story of the game by using visual components while facilitating game edition and understanding. A description based on a narrative metaphor is also being considered in other approaches to educational video game creation (S. Göbel, L. Salvatore, R. Konrad, & F. Mehm, 2008). Moreover, although games could be created

by bundling together interactive puzzles or problems (e.g. this approach is used by the popular *Raptivity*[1] tool), studies show that the narrative structure plays a fundamental role in the understanding of the instructional message and helps new learners to set appropriate goals (D. Laurillard, 1998).

We implemented our approach in order to test it with end-users. Story editing is done through the use of a visual language to represent low-level user interactions (e.g. "grabbing" an object) abstracted as high-level visual components (i.e. an icon in the flow representation). In the WEEV system, simple and explicit visual constructs (i.e. circles, arrows and boxes) replace complex and implicit descriptions (Marchiori, Torrente, Del Blanco, Moreno-Ger, & Fernández-Manjón, 2010). For instance, *conditions*, which result in complex logic structures unfamiliar to non-programmers, are hidden behind the visual components. Moreover, commonly used structures such as multiple-choice questions are included as out-of-the-box and customizable components to fit user needs, instead of being described as complex interrelations between different game elements, thus placing function over interactivity.

This work is structured as follows: In section 2 we present the basic theoretical framework for the WEEV methodology. In section 3 we introduce the WEEV methodology itself. Section 4 provides a description of the WEEV system, the concrete implementation of the WEEV methodology. In section 5 a brief introduction to the related work in the area and a critical analysis of such work is provided. Section 6 presents a formative evaluation, an end-user evaluation and a use case of the system. Finally, section 7 presents the conclusions and future work in this area.

2. Theoretical framework

Educational video games are becoming an increasingly accepted complement to traditional educational approaches. Authors such as Gee (2003), Squire (2004), or Aldrich (2004) argue that games can be used to enhance learning. Amory, Naicker, Vincent, & Adams (1999) argue that computer games could provide a good mechanism to entice learners to acquire knowledge through intrinsic motivation. Computer games can engage students in the learning environment by the use of an interesting story, thus supporting contemporary educational practices (Amory, 2006). Although educational video games also have skeptics and detractors who argue that their efficacy has not been fully demonstrated (Hays, 2005; Pivec, & Pivec, 2008), games have been used in real educational contexts and have achieved good results (Wong, Shen, Nocera, Carriazo, Tang, Bugga, *et al.*, 2007; Coller & Scott, 2009; Blunt, 2007).

The need for an approach to educational game authoring that places educators in an active role is reinforced by End-User Development (EUD) guidelines (Costabile, Fogli, Mussio, & Piccinno, 2007; Fischer, Giaccardi, Ye, Sutcliffe, & Mehandjiev, 2004), which emphasize the need for tools with smooth learning curves targeted at end-users (in this case, educators). The EUD approach tries to reformulate the underlying concepts by using vocabulary and terms that are already familiar to educators and by providing a flow representation, which helps in the design process (Dodero, Martínez del Val, & Torres, 2010).

[1] *http://www.raptivity.com/* (retrieved on July 12, 2011)

The narrative metaphor used to represent the games helps developers to focus on the stories, which serves the educational purpose of games well. Kiili (2005) argues that the story is a very important aspect of game design, as it helps in immersing and engaging the player. According to Dickey (2006), the narrative (i.e. story) in adventure games is useful "both by providing motivation and by serving as a cognitive framework". Dickey (2005) also identifies three elements of interactive design: *setting* to support the narrative and define a *gamespace*; *roles and characters* with the main character (usually the player's avatar) and dialogues establishing the immersion; and *actions, feedback and affordances*, which define what the player can do, as well as victory and loss conditions. Besides, stories provide better support for *intrinsic fantasies*, which Malone (1981) identifies as more interesting and instructional than *extrinsic fantasies*. Tools that support the development of the story can contribute to this, as *intrinsic fantasies* require both problems and feedback to be created within the same fantasy setting. This paper is focused on narrative games, in particular *point-and-click* adventure games that place much emphasis on the story.

Lindley (2005) identifies five semiotic levels (or levels of meaning) in narratives and establishes a relation with the semiotic levels of computer games. The narration becomes the instantiation of an underlying model or simulation in video games, while in traditional media the narration is the instantiation of a story or plot. The model in games usually defines the story implicitly, as a relationship between different elements in the system. The model could, however, be explicitly represented as a story (or at least as the set of all potential stories to be instantiated as narratives by the player). Lindley discusses that approach, although this option is later discarded because it is considered too complex for general game development. The scope of our proposal, however, is limited to educational video games that are exportable as learning objects to be deployed in e-learning systems and of limited length and complexity (Moreno-Ger, Burgos, Sierra, & Fernández-Manjón, 2008; Torrente, del Blanco, Marchiori, Moreno-Ger, & Fernández-Manjón, 2010). Moreover, a hierarchical representation that hides part of the complexity helps to reduce the difficulty of such an approach.

More traditional media (e.g. novels, plays) use narrative or story structures that are repeated across different stories. By considering video games as narrative content, we can use the works of authors such as Dickey (2006) and Lindley (2005) who identified the reuse of the same story structures across video games. These underlying structures can help educators in the development of their video games as they can be reused to provide a guide for the creation of relevant parts of the story and recurring roles in successful narratives. However, the only way for these structures to be reproduced is if they are made explicit and available to developers. This requires the creation of an explicit representation of the game model.

Finally, Dickey (2006) proposes a comprehensive heuristic for game development, based on the consideration of a game as a narrative. This heuristic, which can serve as a guideline and support a structured and systematic approach to educational game development, has the following steps:

- *Present the initial challenge*: the narrative climax becomes the problem or project that is the goal to be learned in an educational context.
- *Identify potential obstacles and develop puzzles, minor challenges, and resources*: smaller obstacles and challenges found in a story become different

procedures, skills and knowledge content that will help learners to complete the challenge in a learning environment.

- *Identify and establish roles*: Using the archetypes identified by Vogler (1998) characters and situations must play certain roles in the game. Most important are the roles of the *hero*, usually performed by the learner, and the role of *mentor*, who provides guidance to foster learner reflection, analysis, planning, and evaluation of strategies.
- *Establish the physical, temporal, environmental, emotional, and ethical dimensions of the environment*: For instance, the physical dimension defines the space in which the player moves. In an educational context, all of these dimensions need to be established to support the storyline by reinforcing plausibility.
- *Create a backstory*: This provides an outline for the different dimensions of the environment, as well as a profile of the protagonist. The main challenge or call to action might be introduced.
- *Develop cut scenes to support the development of the narrative story line*: Cut scenes (i.e. non-interactive game scenes) provide ongoing narrative and may be used to deliver key information or plot hooks. In educational contexts, they could provide feedback about the learner's progress.

3. The WEEV methodology

WEEV is a methodology for the authoring of narrative *point-and-click* educational games. The theoretical works presented in the previous section provide a framework for the definition of the elements in the methodology. In particular, WEEV identifies three main elements or tasks:

1. Definition of the actors: Major actors (i.e. characters and useful objects) that will appear in the story are identified. Actors are described by using detailed descriptions (textual or graphic).
2. Definition of the world: The world where the game takes place is defined to help create an atmosphere (i.e. physical dimension) where the story will develop.
3. Creation of the story: An explicit visual representation of the story is used to describe the flow of the game. The author defines this flow, as well as the puzzles, challenges and story elements that are at the core of the learning process.

The methodology, in contrast to other game authoring approaches, allows for the explicit definition of the elements of interactive design (Dickey, 2005) as shown in Table 1. Every interactive element in the game is defined as an actor (i.e. Non-Player-Characters - NPC - and items), the world is defined as the *gamespace* or virtual game world, and the story is defined as an abstraction of player interactions and the feedback provided by the game. These elements, defined independently, must be edited separately and represent the core of the game definition.

Table 1 Elements of interactive design (Dickey, 2005) and their correlation with elements within the WEEV methodology.

Elements of interactive design (Dickey 2005)	WEEV elements
setting	world
roles and characters	actors
actions, feedback and affordances	story

The different parts of the methodology support the heuristic proposed by Dickey (2006) (Table 2), which encourages a complete and coherent development of educational games and their underlying stories. While the original heuristic presents a theoretical approach to educational game development the WEEV methodology makes practical choices (e.g. abstract concepts are redefined, such as "physical dimension" as the concrete world made of scenes) in other to allow for the direct implementation in an actual development tool. Moreover, even if the heuristic is defined as a set of ordered steps, the WEEV methodology assumes the more realistic scenario where the users need to re-visit parts of the process as required by the development of the different parts of the game.

Table 2 Approximate correlation between the steps in the heuristic proposed by Dickey (2006) and supporting tasks in the WEEV methodology

Heuristic approach (Dickey 2006)	Task in WEEV methodology
Present initial challenge Identify potential obstacles and develop puzzles, minor challenges and resources	Creation of the story
Identify and establish roles	Definition of the actors
Establish the physical, temporal, environmental and emotional, and ethical dimensions of the environment	Definition of the world
Create the backstory Develop cutscenes to support the development of the narrative storyline	Creation of the story

This approach, which divides the game description into interaction elements of narrative significance (i.e. story, world and actors), ensures that the story is represented as a unique element not tangled with programming aspects. This division makes it possible to directly identify the flow of the interactive story, making it easier to find the decision points (divergence in the flow) where students' performance can be assessed (i.e. increase score when making the right choice and decreasing it when making a mistake). The clear identification of decision points also helps by leaving the actual assessment decisions to be made by educators, who can more aptly identify those of educational interest.

In conclusion, the use of this methodology is intended to provide different benefits for the game developer, as it is focused on the game's story and uses concepts that are not

related to programming. The structured approach supported by the WEEV methodology provides a framework that explicitly addresses problems that are recurrent in video game creation such as integration of the narrative with the puzzles and challenges. However, an excessively rigid application of the methodology could result in limitations in the creation of long games because not all elements in the game (e.g. art resources) are easily defined within the methodology. These advantages and problems were taken into account in the implementation of the WEEV methodology into the WEEV system.

4. The WEEV system

The WEEV system is the implementation of the WEEV methodology into an actual game creation framework. The WEEV system is built upon <e-Adventure>[2], an educational video game authoring platform created with the aim of enhancing the teacher's implication in the game development process by reducing the related cost and technical requirements while increasing the created game's educational value (Moreno-Ger, *et al.*, 2008; Torrente, *et al.*, 2010). This platform has been used in different educational settings (Moreno-Ger, Torrente, Bustamante, Fernández-Galaz, Fernández-Manjón, & Comas-Rengifo, 2010) and uses a content-centric approach to video game creation.

<e-Adventure> supports educational features such as the assessment of the student's performance (i.e. by creating detailed reports of student interaction) or the adaptation of the game experience in order to adjust the game to different students' needs (e.g. showing puzzles with different difficulties depending on the student's past grades). The integration with this platform ensures that the games can be exported to any of the formats supported by <e-Adventure>. This includes creating Learning Objects from the game, executable files, or files that can be further edited by using the editor tool in the <e-Adventure> platform. Allowing games to be further edited in <e-Adventure> removes limits imposed by the high-level abstraction used in WEEV, making possible the direct implementation of new features and behaviors by expert users (this might, however, result in the game being no longer editable in WEEV).

WEEV is implemented as three tools to edit the main elements in the methodology: a simple actor editor, a world editor that uses a Domain Specific Visual Language (DSVL) and a story editor that uses another DSVL. The use of DSVL is intended to provide an easy-to-use and understandable representation of the most complex elements in the game (Boshernitsan, & Downes, 2004; Marriott, Meyer, & Wittenburg, 1999). The world representation includes the complete description of the virtual game world, while the story description abstracts user/system interactions as high-level visual components.

4.1 Definition of actors

Every interactive element in the story (i.e. every element upon which the player can perform an action) is defined in the WEEV methodology as an actor. These actors are edited in a single list, regardless of their function or representation in the game, which can vary from non-player characters to game objects (i.e. interactive elements the player may find throughout the story). Actors in the WEEV implementation can be edited in a

[2] *http://e-adventure.e-ucm.es* (retrieved on July 12, 2011)

specific panel, which presents them in a simple list with all the relevant properties and values.

When a new actor is added a "resource library" is offered to select an appropriate set of graphic resources. However, while resource libraries facilitate simple game development, the creation of an extensive library of graphic resources to cover most usual game situations is an expensive task, and even the most comprehensive libraries end up being limiting. For this reason, users can also choose to use their own custom resources.

4.2 World edition

The edition of the game world uses a DSVL that allows for the definition of *spaces* (Figure 1, a). *Spaces* are places that the player may visit during the game and where the game action takes place. *Spaces* can be linked with other spaces, creating *spacelinks* that allow the player to go from one *space* to another (Figure 1, b). The ability of the player to use a *spacelink* can later be constrained in the story definition; for instance, the "Go to kitchen" link exists, but the player may not use it when a door is still locked in the story.

Moreover, actors in the game can be placed in one or more *spaces* in the world. The *actor placement* representation elements show the distribution of the actors (including NPC and items) in the different *spaces* of the game (Figure 1, c).

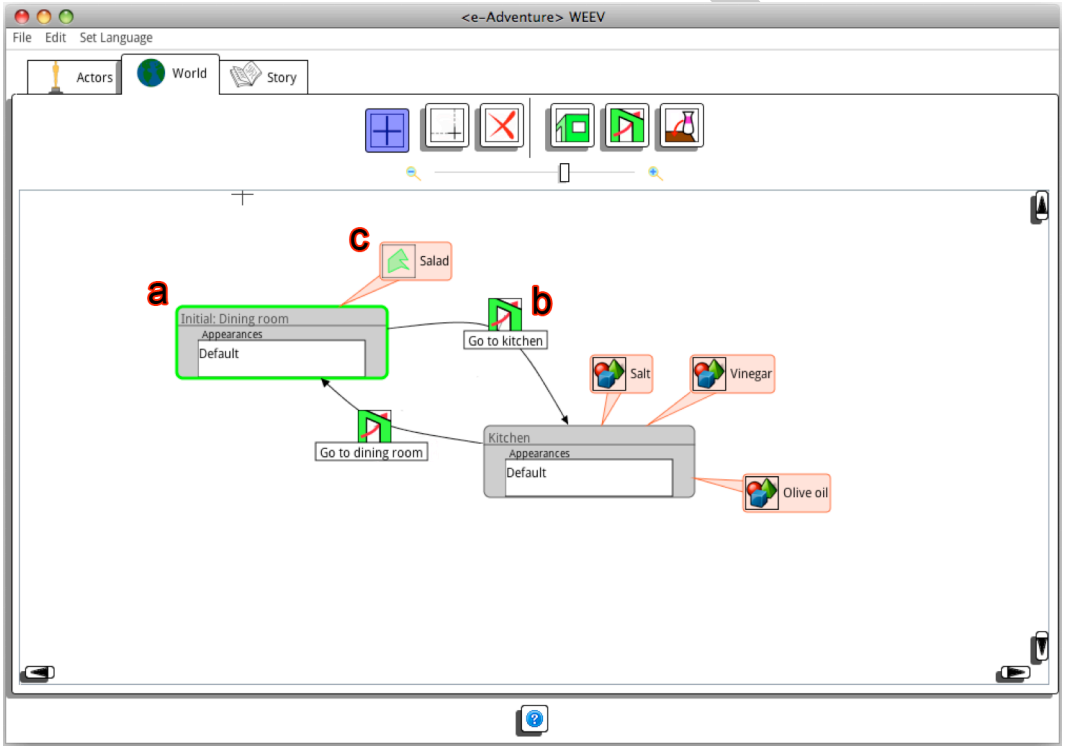

Figure 1 The world edition view in the WEEV system. This screenshot shows different *spaces* (a), *spacelinks* (b) and *actor placement* elements (c).

Spaces can also have different appearances assigned to them. These appearances will allow the creator to define a change in the physical aspect of a *space* from the player's perspective during the game. These changes in appearance are determined by the game's story. Graphic resources are defined through the "resource library" or by using custom assets.

4.3 Story edition

For story edition, a comprehensive DSVL was defined (Marchiori, E. J., Del Blanco, A., Torrente, J., Martinez-Ortiz, I., & Fernández-Manjón, B., 2011). This DSVL is defined in a similar way to a visual programming language (Boshernitsan, *et al.*, 2004; Marriott, *et al.*, 1999) and can be directly transformed into a playable game. It allows for a hierarchical representation of the information, which encapsulates parts of the story, making it easier to understand.

The representation of the story is based on the explicit representation of the interactions between the user and the game. This representation is based on a state-transition diagram, where each state represents a point in the game story and each transition an interaction by the user with the system, which moves the story along. This abstraction allows for a comprehensive representation of the full story of the game, including system feedback and adaptation (i.e. changing the game depending on external variables). Moreover, this representation can be used for other simple games such as procedural simulations (i.e. simulations of simple procedures with pre-defined consequences), where the story is less defined but the procedure can be represented as a state-transition diagram as well.

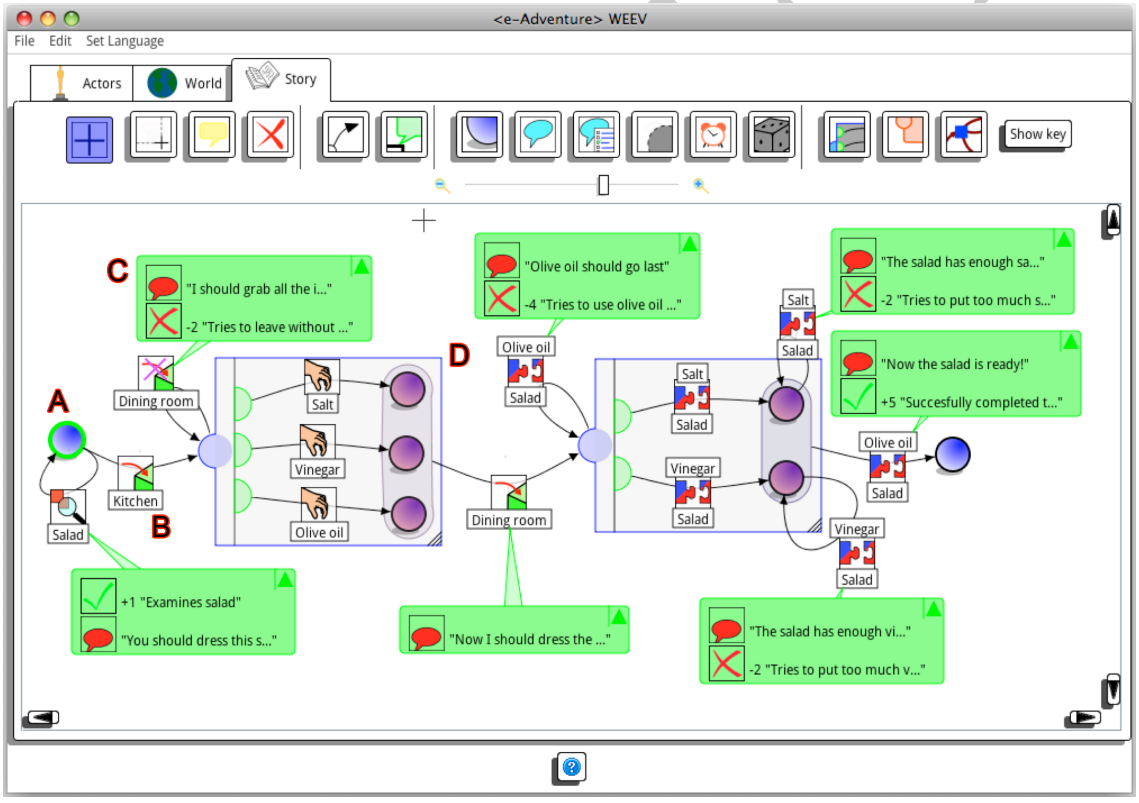

Figure 2 The story edition panel presents a toolbar on top to add new expressive elements to the story as well as a representation of the story itself. In this case, the game proposed in the formative evaluation of WEEV is represented by the different elements of the DSVL, which include A. Game states, B. User actions, C. Effects (e.g. showing text and assessing the user behavior) and D. Multi-interactions (i.e. two or three things that can be done in any order but are needed in order for the story to move along).

Other language constructs cover complex user interactions (e.g. out-of-order interactions), and other specific elements (e.g. multiple-choice questions). The DSVL helps to understand the game story better if someone else created it, which increases game maintenance, facilitates co-authorship and allows educators to add educational value to existing games, increasing their reusability and usefulness. Besides, the use of visual languages has been shown to enhance the understanding and access to relevant information (Navarro-Prieto, & Cañas, 2001). A special edition panel is used, with tools to add the different elements in the language to the story (Figure 2).

This language must be distinguished from other visual languages used in educational contexts to express sequences, such as those found in *LAMS*[3]. In this case the DSVL describes the game story through low-level user interactions (e.g. answering certain question) rather than whole lectures or courses through sequences of activities (e.g. answering all the questions in a self-evaluation test). However, the use of such languages in other educational contexts might help new users to the WEEV system who are already familiar with the basic visual language concepts.

The language also includes educational features such as student evaluation. The underlying evaluation engine in <e-Adventure> is used for this, providing both educators and students with a detailed account of relevant interactions from an educational point of view. In the same manner, the language allows the use of the adaptation engine in <e-Adventure> to provide slightly customized experiences of the game to different users (e.g. different questions for users with different levels of knowledge of the subject).

4.4 Using the system

Current implementations of the WEEV system integrates the three distinct tools but distributed in separate tabs for easy access, one for each element in the methodology (actors, world and story, the last two as shown in Figure 1 and Figure 2). When starting a new project (Figure 3, a) the user is presented with a blank game, where there is just a main actor (i.e. the player), a space (i.e. the initial space) and an empty story. In the same way, the user can load a previously saved game project (Figure 3, b). Although a typical game creation session starts with the definition of the actors, followed by the definition of the spaces and the creation of the story, the use of tabs makes each of these elements easily editable at any point in the development, allowing the user to go from one to the other as needed (Figure 3, c).

As games implemented with the WEEV system can be directly translated to <e-Adventure> games (Figure 3, d), WEEV provides features that directly run games and see the story unfold as defined in the DSVL, convert games as <e-Adventure> projects that can be further edited, modified and improved (Figure 3, e), and finally, export games in different formats that can be included in Learning Management Systems (LMS), played directly in a PC or included in a website (i.e. Java applets).

[3] *http://lamsfoundation.org/* (retrieved on July 12, 2011)

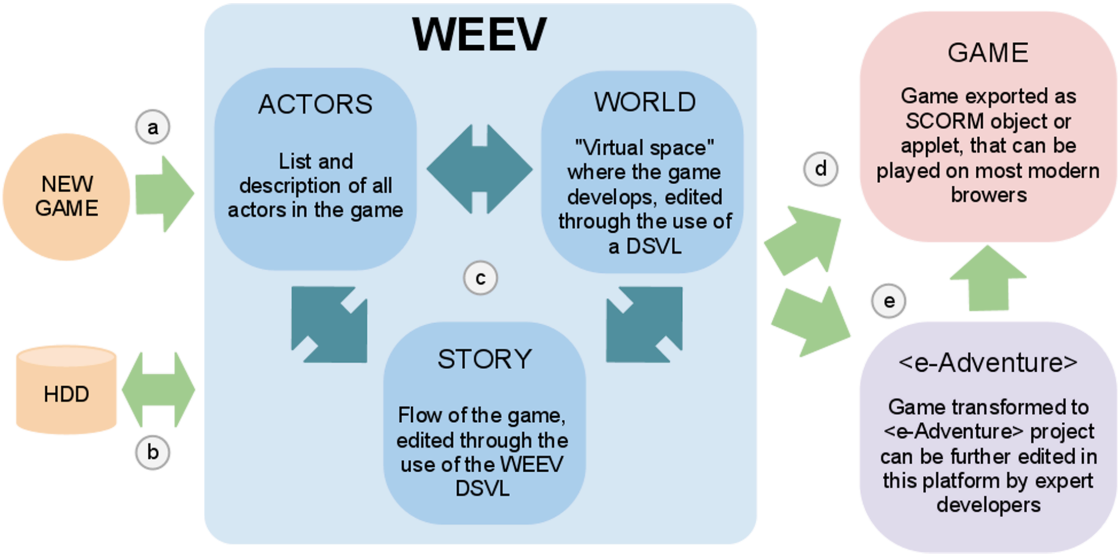

Figure 3 The WEEV edition process goes from new or previously saved games to the creation of actual games or <e-Adventure> projects that can be edited further. Within the WEEV system the user can edit actors, world and story in any order and going from one edition mode to the others.

5. Related work

Other systems attempt to simplify the creation of games or introduce games in education. While some of these systems share particular features with WEEV, others take very different approaches. This section will cover only some systems that we consider most representative to provide a better framing of our contribution, with Table 3 summarizing the main features of each system in comparison with WEEV. A more detailed discussion of the creation of educational games, including the costs and tools involved, is available in (Wexler, Corti, Derryberry, Quinn, & Barneveld, 2008).

Thinking Worlds[4] is a widely-used commercial tool that facilitates the creation of serious 3D games. It also attempts to simplify educational video game development by using an explicit representation of the game flow, although it uses a hybrid visual language with a strong "textual" component. *Thinking Worlds* includes several tutorials for novice users but it is targeted towards the creation of more complex 3D games (e.g. it requires camera placement, creation of paths for the camera as well as characters through the scene, etc.), which could result in a steeper learning curve. Games created with this system can be played through a web browser and packaged as SCORM objects to be included in LMS.

Adventure Author (Robertson, & Good., 2005; Robertson, & Nicholson, 2007) focuses on young learners as the target group and uses a set of plug-ins to create games for the *Neverwinter Nights 2* commercial game engine. This system is good at encouraging young students to create stories and is very easy to use. The approach is based on the visual language similar to the one in WEEV, but focused on the development of linear stories (i.e. no real interactivity) rather than games. Besides, it does not include educational features, making, for instance, external assessment of in-game performance a requirement.

---

4 *http://www.thinkingworlds.com/* (retrieved on July 12, 2011)

Similarly, *Storytec* (S. Göbel, *et al.*, 2008) proposes the use of another visual language for the creation of serious games. Just like *Adventure Author*, it is targeted more to the creation of non-linear stories to encourage creativity than to the creation of educational games. And just like *Adventure Author*, it suffers from a lack of educational features necessary to assess performance. While *Storytec* provides powerful expressiveness for story creation, it uses a complex set of views that increase the learning curve.

Storytelling Alice (Kelleher, & Pausch, 2007) and *Alice*[5] were created to teach programming concepts to young children and provide a graphic language that allows for the creation of interactive stories. These systems have very smooth learning curves and although they are not specifically devised for educational applications (i.e. no assessment, limited interactivity), they are widely used in educational settings due to their simplicity and polished design. These systems are usually used directly by the students, rather than the resulting games created with them.

Many other commercial and free game development platforms are available, such as *Adventure Game Studio*[6], *Adventure Maker*[7] and *Unity*[8]. These systems can potentially be used to create educational games, although they are not targeted to educators, do not provide tools for their direct integration with LMSs and have steep learning curves. In addition, none of them have explicit educational features to allow assessment of student performance or the measurement of other criteria relevant in education. They all provide graphical (GUI) editors, although they require the user to learn programming languages of varying complexity.

Table 3 Summary of WEEV features and their equivalents in similar systems

Platform	Licensing	Main approach	Educational features	Standards	Edition complexity
WEEV	LGPL	Visual Language	Evaluation, Adaptation	SCORM, IMS CP	Low/High
<e-Adventure>	LGPL	GUI Editor	Evaluation, Adaptation	SCORM, IMS CP	High
Thinking Worlds	Commercial	Hybrid Visual Language	Evaluation	SCORM	Low
Adventure Author	NA	GUI Editor	NA	NA	High
Storytec	NA	Visual Language, undefined	NA	NA	NA
Storytelling Alice	Free	Visual Language	NA	NA	Low
Adventure Game	Free	GUI Editor	NA	NA	High

5 *http://www.alice.org/* (retrieved on July 12, 2011)

6 *http://www.adventuregamestudio.co.uk/* (retrieved on July 12, 2011)

7 *http://www.adventuremaker.com/* (retrieved on July 12, 2011)

8 *http://unity3d.com/* (retrieved on July 12, 2011)

Studio					
Adventure Maker	Free, commercial	GUI Editor	NA	NA	High
Unity	Commercial	GUI Editor	NA	NA	Very high

6. Evaluation

Three different evaluations were performed on the system. The first evaluation was of a formative nature, intended to improve the software. The second evaluation, performed after the improvements to the software were implemented, was performed with end-users (i.e. educators) to assess the potential of the approach. The third evaluation consisted of the creation of an actual game by using the system.

The stories or games used in the first two evaluations have no educational value, because the educators are the ones expected to introduce it while creating their own games (or improving upon existing ones). During these evaluations the goal was to successfully turn simple stories into working games, for which simple and clear stories were selected over others that might have had direct educational use. At the same time, educators are expected to have an advisory role in most cases (as shown in the third evaluation), but the first evaluations were intended to test the deep understanding of the metaphor that would allow such a role to be possible.

6.1 Formative evaluation

The goal of the formative evaluation was to allow for the early discovery of problems in the methodology, the software and user interaction with the system. These evaluations are usually performed with users familiar with the subject matter. To better achieve our goal, we selected 20 software engineering students taking a seminar on educational video game development at our university.

The evaluation was performed in a laboratory, where a brief introduction to the system was provided (less than 5 minutes). A simple game was proposed to the students (including the graphic assets) and they were given around 50 minutes to implement it. After the evaluation, a survey and informal questions were used to assess users' impression about the system. During the session, users were encouraged to report technical errors (e.g. error messages), problems with the interaction (e.g. non-responsive clicks), or any other implementation concerns (e.g. difficulty operating menus).

The users had enough time to attempt to implement the simple story, one possible implementation of which is shown in Figure 2. During this time some users (7 of them) detected important problems in the software (e.g. some saved files were unreadable or some constructs needed to be deleted in order to be modified), none of which stopped them from being able to use the system successfully. This information was directly applied to improved successive versions of the system.

Users found the information provided by the system excessive in some cases (i.e. help panels provided lots of information of limited relevance) while other interactions with the system failed to inform why some actions were prohibited. This provided clues as to how to improve the quality and quantity of information for future versions, while at the same time helped us realize that parts of the system required more information.

On a negative note, the students participating in this evaluation did not seem to value the usefulness of the software as much as expected. In our opinion, this could be due to the fact that all of them were familiar with programming languages and found no need for a graphic and explicit representation of the flow, and also because we failed to successfully explain the goals of the software. We tried to improve upon this aspect in the following evaluations.

6.2 End-user evaluation

The goal of this evaluation was to assess the impression and perception of the system by educators. Nine members from the department of Pedagogy at the Complutense University of Madrid were invited. These users showed interest in the use of educational video games in learning scenarios and also in technologies that would allow them to create they own games, but lacked the required programming skills to create a game on their own with existing tools.

In this case we used a guided approach, where the users were asked to perform different tasks and given time to complete them on their own. The goal of the session was to recreate the story of "Little Red Riding Hood" as an interactive game, with the player assuming the role of the girl in the story. This was chosen for its familiarity and because it presents two divergent paths (i.e. listening to the wolf or distrusting him), which are included in sequence in the original story. This divergence of paths creates non-linearity in the story, making it more interesting as a game. Through this story we intended to establish if the underlying methodology (and not necessarily the system) was correctly interpreted by, and useful to, educators.

This evaluation lasted around 90 minutes, but the users did not have time to fully develop the story. This was due to the fact that the guided approach is much slower and that the users were unfamiliar with the subject and required more time to perform the tasks required. Still, all the users had enough time to recreate parts of the story, run them as games, test and modify them and try them again, allowing them to correctly understand the metaphor and the goal of the system.

The users found the biggest problem to be the complexity of the tool, because even if they had found the explanations satisfactory and the metaphor useful and understandable, they had many usability problems (e.g. the system was complex to use, especially at the creation of new elements). This prompted us to consider a reimplementation of the story edition interaction (i.e. how the tool is presented to the user) using a different paradigm for future versions.

The main conclusions of this evaluation were the following:

- The users stated a clear interest in using the tool for their personal projects, as they felt comfortable enough to try to develop their own stories
- The ability to test games as they were being developed, seeing the changes in the visual language reflected in the actual game, proved fundamental in helping the educators understand the metaphor

The educators also provided some useful suggestions from their point of view (and that of other non-technical users):

- Educators needed example games to better understand the purpose of the system (similar to the conclusion arrived at by Ketelhut & Schifter (2011), regarding the need to provide teachers with models of successful implementation). We believe

this need arises mostly out of the novelty of the approach, which introduces an interaction metaphor with the system that is unfamiliar to educators.

- The inclusion of a guided tutorial within the tool would help users during the first contact with the tool, which can possibly be done by an approach similar to the one found in *Storytelling Alice* (Kelleher, & Pausch, 2005).

6.3 Creating an actual game

After the first two evaluations the authors had the opportunity to create a series of games for CATEDU[9] (a Spanish center for the dissemination of educational technology). We selected one of the games, dealing with learning English, to be developed using the WEEV system, and asked the English teacher involved to take an active role. This game was defined as a short (15 minutes, approximately) story-based game.

The game represented the main player on a business trip to London, with the intention of providing context for the learning of English vocabulary and grammar by secondary school students. Using the WEEV system allowed for the incremental development of the game, where the educator was able to provide useful feedback both on the content and the flow by understanding the inner workings of the game (Figure 4). The final game has over 30 vocabulary questions and develops over 7 different scenarios (e.g. office, airport).

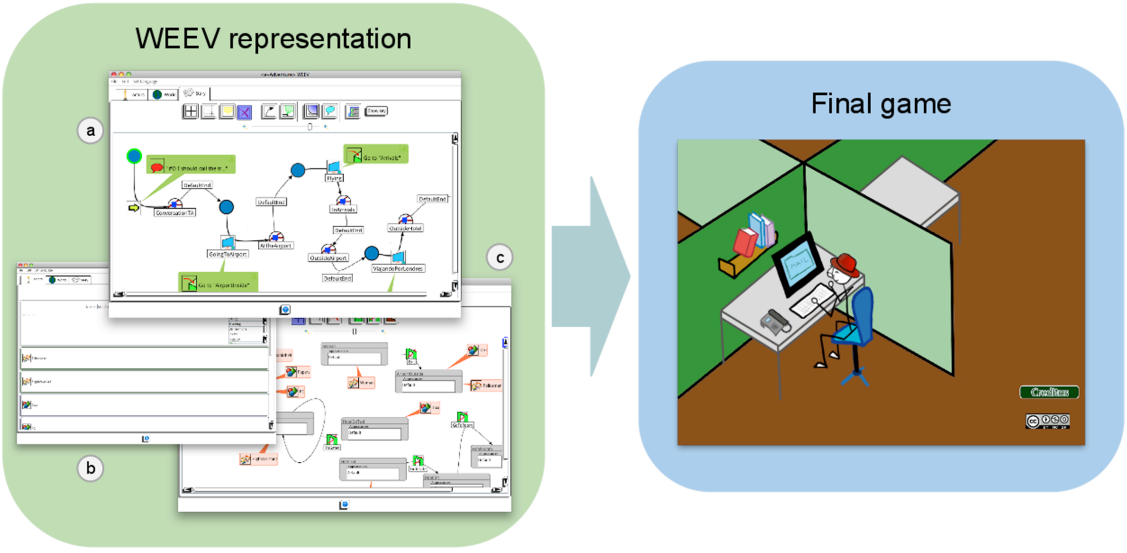

Figure 4 A story created using the story (a), actors (b) and world (c) editors in WEEV was converted into a functional educational game

Currently the game is freely available on-line and is being distributed to different education centers. We expect to be able to collect additional information from the experience of using this game in the centers, mainly to validate the pedagogical approach of the game (i.e. simple adventure game with heavy reliance on multiple-choice questions).

9 *http://www.catedu.es* (in Spanish, retrieved on July 12, 2011)

The approach was useful for the following reasons:

- The final game was almost entirely developed within the WEEV system (with the exception of a mini-game that was added later on).
- It was possible (and easy) to incorporate the educator into the development process thanks to the explicit representation of the game-flow.
- The explicit assessment representation in WEEV allowed us to easily define the consequences of wrong and right answers to different questions in the game.

However, this first real application of the system allowed us to detect different limiting factors that either were addressed during the development or should be addressed in future versions:

- Many structures are repeated within games and across different games, and as such they should be easy to include in games. We added a multiple-choice question component to the system, but the possibility to add mini-games directly into the game-flow is still in development.
- To provide true integration of multiple team members, an online collaboration approach should be included as a part of the system.

7. Conclusions and Future work

This paper presented the WEEV methodology and system. WEEV's main goal is to facilitate educational video game development, while making it easy to include educators in the development process. Involving educators is the best way to ensure a high degree of usefulness and educational value within the games. To achieve this goal, the WEEV methodology is grounded on video game and narrative theory, while the system uses a set of visual languages to create explicit representations of the elements of interactive design. The WEEV system allows for the direct implementation of games following the WEEV methodology, creating games that are described by a narrative metaphor.

WEEV was subjected to a formative evaluation with advanced users and an end-user evaluation with educators. The results of the evaluation show that in general users appreciate and understand the methodology and the use of a narrative metaphor to represent game stories. The current system implementation is somewhat difficult to use for some users (especially those with more limited technical knowledge), partly because of the visual complexity of the editor (i.e. many different tools, most of them mutually exclusive, are always available in the toolbar). Moreover, a case of actual use of the WEEV system in the development of an educational game currently being used in high-school English courses allowed us to further test the approach, confirming its benefits and finding some limitations that will need to be addressed.

We find that the overall results of the evaluations are promising, especially regarding the methodology. The explicit representation of the game-flow makes it easier both to develop and understand the game, and to describe the way the game works for non-expert users, facilitating that educators will add educational value to existing games (e.g. student performance assessment dependant on the actions in the game). Besides, this is reinforced by the visual language, which allows the identification of the most important points in the story. Although some (mostly technical) aspects need further work, the system is ready to be tested and applied in other scenarios.

The next step in the project is to examine and include the helpful feedback gathered from educators in order to improve the system, while changing the interaction paradigm from a modal edition (i.e. the use of a complex toolbox as in advanced tools like Adobe Photoshop) to a modeless edition (i.e. the use of just contextual options that relate to the currently selected elements in the edition with explicit effects) that should reduce complexity. The system will then undergo a new evaluation process involving more educators with different backgrounds and experience. This process includes using quantitative measures to assess the use of the system and qualitative observations of user behavior and opinions. Our future work also evaluates WEEV as a means to foster young learners' creativity, such as the use given to *Adventure Author* and *Storytelling Alice*.

Acknowledgments

The Ministry of Education (grants Movilidad I-D+i PR2010-0070 and TIN2010-21735-C02-02) and the Ministry of Industry (grants TSI-020110-2009-170, TSI-020312-2009-27) have partially supported this work, as well as the Complutense University of Madrid and the Regional Government of Madrid (research group GR35/10-A and project e-Madrid S2009/TIC-1650), and the PROACTIVE EU project (505469-2009-LLP-ES-KA3-KA3MP) and the GALA EU Network of Excellence in serious games (FP7-ICT-2009-5- 258169).

We would also like to thank all those that willingly provided their time, effort and insight by freely participating in our evaluations, and Tina Christodouleas, PhD and Scott C. Goodell for their detailed review of the manuscript.

References

Aldrich, C. (2004). *Simulations and the future of learning: An innovative (and perhaps revolutionary) approach to e-learning*. San Francisco, CA: Pfeiffer.

Amory, A. (2006). Game object model version II: a theoretical framework for educational game development. *Educational Technology Research and Development,* 55(1), 51-77.

Amory, A., Naicker, K., Vincent, J., & Adams, C. (1999). The use of computer games as an educational tool: Identification of appropriate game types and game elements. *British Journal of Educational Technology*, 30(4), 311-321.

Blunt, R. (2007). Does game-based learning work? Results from three recent studies. In proceedings of: *Interservice/Industry Training, Simulation & Education Conference (I/ITSEC) (pp.* 945-954). NTSA, Orlando, Florida, USA.

Boshernitsan, M., & Downes, M. (2004). *Visual programming languages: A survey*. Berkley, California: University of California. <http://www.eecs.berkeley.edu/Pubs/TechRpts/2004/CSD-04-1368.pdf >

Bourgonjon, J., Valcke, M., Soetaert, R., & Schellens, T. (2010). Student's perceptions about the use of video games in the classroom. *Computers & Education*, 54(2), 1145-1156.

Coller, B.D., & Scott, M.J. (2009). Effectiveness of using a video game to teach a course in mechanical engineering. *Computers & Education*, 53(3), 900-912.

Costabile, M. F., Fogli, D., Mussio, P., & Piccinno, A. (2007). Visual interactive systems for end-user development: A model-based design methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, 37(6), 1029-1046.

Dickey, M. D. (2005). Engaging by design: How engagement strategies in popular computer and video games can inform instructional design. *Educational Technology Research and Development*, 53(2), 67-83.
Dickey, M. D. (2006). Game design narrative for learning: Appropriating adventure game design narrative devices and techniques for the design of interactive learning environments. *Educational Technology Research and Development*, 54(3), 245-263.
Dodero, J. M., Martínez del Val, Á., & Torres, J. (2010). An extensible approach to visually editing adaptive learning activities and designs based on services. *Journal of Visual Languages and Computing*, 21(6), 332-346.
Fischer, G., Giaccardi, E., Ye, Y., Sutcliffe, A. G., & Mehandjiev, N. (2004). Meta-design: A manifesto for end-user development. *Communications of the ACM*, 47(9), 33-37.
Gee, J. P. (2003). *What video games have to teach us about learning and literacy*. Palgrave Macmillan, New York; Basingstoke.
Göbel, S., Salvatore, L., Konrad, R. A., & Mehm, F. (2008). StoryTec: A digital storytelling platform for the authoring and experiencing of interactive and non-linear stories. In: U. Spierling, & N. Szilas, *Interactive Storytelling (Vol. 5334, pp. 325-328)*. Berlin / Heidelberg: Springer.
Hays, R. T. (2005). *The effectiveness of instructional games: a literature review and discussion*. Naval Air Warfare Center, Orlando, FL. <http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA441935>
Kelleher, C., & Pausch, R. (2005). Stencils-based tutorials: design and evaluation. In proceedings of: *Conference on Human Factors in Computing Systems (pp. 541-550),* Portland, Oregon, USA.
Kelleher, C., & Pausch, R. (2007). Using storytelling to motivate programming. *Communications of the ACM*, 50(7), 58-64.
Ketelhut, D. J., & Schifter, C. C. (2011). Teachers and game-based learning: Improving understanding of how to increase efficacy of adoption. *Computers & Education*, 56(2), 539-546.
Kiili, K. (2005). Digital game-based learning: Towards an experiential gaming model. *Internet and Higher Education*, 8(1), 13-24.
Laurillard, D. (1998). Multimedia and the learner's experience of narrative. *Computers & Education*, 31(2), 229-242.
Lindley, C. A. (2005). Story and narrative structures in computer games. In: B. Bushoff, *Developing interactive narrative content*. Munich: Sagas/Sagasnet reader.
Malone, T. (1981). Toward a theory of intrinsically motivating instruction. *Cognitive Science*, 5(4), 333-369.
Marchiori, E. J., Torrente, J., Del Blanco, A., Moreno-Ger, P., & Fernández-Manjón, B. (2010). A visual domain specific language for the creation of educational video games. *IEEE Learning Technology Newsletter*, 12(1), 36-39.
Marchiori, E. J., Del Blanco, A., Torrente, J., Martinez-Ortiz, I., & Fernández-Manjón, B. (2011). A visual language for the creation of narrative educational games. *Journal of Visual Languages and Computing*, In press (DOI: 10.1016/j.jvlc.2011.09.001).
Marriott, K., Meyer, B., & Wittenburg, K. B. A. (1999). Survey of visual language specification and recognition. In: Marriot, K., Meyer, B., *Visual Language Theory*. New York, NY: Springer-Verlag.
Moreno-Ger, P., Burgos, D., Sierra, J. L., & Fernández-Manjón, B. (2008). Educational game design for online education. *Computers in Human Behavior*, 24(6), 2530-2540.

Moreno-Ger, P., Torrente, J., Bustamante, J., Fernández-Galaz, C., Fernández-Manjón, B., & Comas-Rengifo, M. D. (2010). Application of a low-cost web-based simulation to improve students' practical skills in medical education. *International Journal Medical Informatics*, 79(6), 459-467.
Navarro-Prieto, R., & Cañas, J. J. (2001). Are visual programming languages better? The role of imagery in program comprehension. *International Journal of Human-Computer Studies*, 54(6), 799-829.
Pivec, M., & Pivec, P. (2008). Games in schools. In: ISFE-EUN Partnership, *Work Package 2: Games in Schools*. <http://www.paulpivec.com/Games_in_Schools.pdf>
Robertson, J., & Good, J. (2005). Adventure Author: An authoring tool for 3D virtual reality story construction. In proceedings of: *Artificial Intelligence in Education (AIED), Workshop on Narrative Learning Environments*, Amsterndam, Netherlands.
Robertson, J., & Nicholson, K. (2007). Adventure Author: a learning environment to support creative design. In proceedings of: *Sixth International Conference for Interaction Design and Children (IDC) (pp. 37-44)*, Aalborg, Denmark.
Ryan, M. L. (2006). *Avatars of story*. Minnesota, USA: University of Minnesota Press.
Squire, K., & Barab, S. (2004). Replaying history: Engaging urban underserved students in learning world history through computer simulation games. In proceedings of: *Sixth International Conference of the Learning Sciences (pp. 505-512)*, Santa Monica, United States.
Torrente, J., del Blanco, Á., Marchiori, E. J., Moreno-Ger, P., & Fernández-Manjón, B. (2010). <e-Adventure>: Introducing educational games in the learning process. In proceedings of: *IEEE Engineering Education Conference (EDUCON) (pp. 1121-1126),* Madrid, Spain.
Vogler, C. (1998). *The writer's journey: Mythic structures for writers*. Studio City, CA: Michael Wiese Productions.
Wexler, S., Corti, K., Derryberry, A., Quinn, C., & Barneveld, A. V. (2008). *The eLearning Guild: 360 report on immersive learning simulations*. <http://www.elearningguild.com/research/archives/index.cfm?id=121>
Wong, W. L., Shen, C., Nocera, L., Carriazo, E., Tang, F., Bugga, S., Narayanan, H., Wang, H., & Ritterfeld, U. (2007). Serious video game effectiveness. In proceedings of: *ACM International Conference on Advances in Computer Entertainment Technology (pp. 49-55)*. Salzburg, Austria.