Interactive Story Authoring: A Viable Form of Creative Expression for the Classroom

Mike Carbonaro[a], Maria Cutumisu[b], Harvey Duff[a], Stephanie Gillis[c], Curtis Onuczko[b], Jeff Siegel[b], Jonathan Schaeffer[b], Allan Schumacher[b], Duane Szafron[b] and Kevin Waugh[b]

[a]*Faculty of Education, University of Alberta, Edmonton, AB, Canada, T6G 2G5*

[b]*Department of Computing Science, University of Alberta, Edmonton, AB, Canada, T6G 2E8*

[c]*Edmonton Catholic Schools, 9807 – 106 Street, Edmonton, AB, Canada, T5K 1C2*

Abstract

The unprecedented growth in numbers of children playing computer games has stimulated discussion and research regarding what, if any, educational value these games have for teaching and learning. The research on this topic has primarily focused on children as players of computer games rather than builders/constructors of computer games. Recently, several game companies, such as BioWare Corp. and Bethesda Softworks, have released game story creation tools to the public, along with their games. However, a major obstacle to using these commercial tools is the level of programming experience required to create interactive game stories. In this paper, we demonstrate that a commercial game story construction tool, BioWare Corp.'s Aurora Toolset, can be augmented by our new tool, ScriptEase, to enable students in two grade ten English classes to successfully construct interactive game stories. We present evidence that describes the relationship between interactive story authoring and traditional story authoring, along with a series of factors that can potentially affect success at these activities: gender, creativity, intellectual ability, previous experiences with programming, time playing computer games, and time spent online. Results indicate that students can successfully construct sophisticated interactive stories with very little training. The results also show no gender differences in the quality of these interactive stories, regardless of programming experience or the amount of time per week playing computer games or participating in general online activities, although a subset of female students did show a slightly higher level of performance on interactive story authoring. In the educational context of this study, we show that ScriptEase provides an easy-to-use tool for interactive story authoring in a constructionist learning environment.

Keywords: Interactive story authoring, scripting, role-playing games, Neverwinter Nights.

1. Introduction

Computer and video games are rapidly becoming omnipresent in our culture, especially in the lives of many children and adolescents (Rhyne, 2002). The research debate continues as to the educational value of these types of games (Mitchell & Savill-Smith, 2004). On the one hand, a recent study reported that video games are a good workout for the brain and that video game players actually perform faster on certain language tasks than their non-game playing counterparts (Bialystok, 2006). Gee (2003) suggests that computer/video games are a powerful educational medium that promote a new form of literacy. He argues that computer game playing encourages the development of problem-solving behavior, motivation and social networks, and that it also enhances and supports learning. On the other hand, Mori (2002) provides evidence that prolonged periods of playing computer games can lead to decreased brain activity and may result in emotional and behavioral problems. A number of researchers have reported that computer game play is directly related to increased levels of aggression or even violent behavior (Anderson & Bushman, 2001). The issues surrounding the educational value of computer games are further complicated when gender is considered, since girls often exhibit less positive attitudes toward using technology (Young, 2000). Studies also indicate that girls are more interested in games of skill than games of kill (AAUW,

2000; Yelland & Lloyd, 2001). Furthermore, it has been shown that boys spend significantly more time playing computer games than girls (Bryce & Rutter, 2003; Griffiths & Hunt, 1995).

There is little doubt that debate concerning use of computer games in the educational context will continue as the number of hours children and adolescents spend playing games increases. What underlies almost all the research in this area is that users are studied as "players of games" and not as "builders or designers" of games (Robertson & Good, 2005; Mitchell & Savill-Smith, 2004). In this study, we examine the role of high school students as authors (designers and builders) of interactive computer game stories. In this context, we offer a new computer game construction medium for creative authoring that allows a student to build interactive stories where the "reader" is an active participant in the story. We refer to this process as interactive story authoring. A major component of support for this new medium is the development of a high-level software tool called ScriptEase (http://www.cs.ualberta.ca/~script/) that alleviates the need for explicit computer programming during the game story authoring process.

A classroom transition from observing ubiquitous artifacts for learning purposes to creating artifacts as a form of creative expression has occurred to various degrees for many media. For example, visual arts such as drawing and painting have made an easy transition in the classroom from observing (viewing) to creating. Sculpture though less prevalent as a creative medium in the classroom is not rare. In fact, today's classroom probably has more activities for creating visual arts than viewing them. A similar phenomenon has occurred in the audio arts for both orchestral and choral music. However, for audio arts the current norm falls between observation and creation. Students tend to participate in activities that interpret the creative works of others by playing and singing music, rather than listening or writing music. Unfortunately, more complex media that combine visual and audio components, such as plays and movies, are much more popular as observational media than as creative media in today's classrooms. When a play is performed rather than read, proper classroom instruction may transform the performance activity from an observational activity to an interpretive activity, but little play-writing or movie-making is done in the classroom. The situation with interactive stories and computer games is similar. Although they may be used in some classrooms as observational or interpretational activities, they are rarely used as a medium for creative expression. The reason for this is simple: the technology required for authoring has been too hard to use.

The research described in this paper makes five contributions that enable computer game authoring to become a viable mechanism for interactive story authoring in the classroom. First, it identifies the technology required for students to create interactive stories in the classroom, along with the current major barrier to viability – manual scripting (computer programming). Second, it presents a solution to this problem – generative patterns that create program scripts automatically. This solution is embedded in our easy-to-use ScriptEase tool. These first two contributions appear in Section 4. Third, it presents the results of a case study showing that grade ten high school English students can use ScriptEase to overcome the scripting problem and create interactive stories. Fourth, it provides some evidence that a segment of students who do poorly relative to their peers on traditional writing exercises can exhibit relative improvement in interactive story authoring. Fifth, it provides evidence to indicate that students are not disadvantaged in their ability to write interactive stories using the computer game development medium, regardless of a) gender, b) creative ability, c) general intellectual ability, d) the amount of prior computer game playing experience, and e) the amount of prior experience using the Web for surfing and chatting. Contributions three to five are presented in Sections 5 and 6. Interactive story authoring technologies offer students the opportunity for a new form of creative expression that supports alternative narrative development outside the realm of text or speech.

2. Learners authoring their own game stories

The ability to design and create game stories has essentially been left in the hands of commercial development teams consisting of programmers, writers, artists, and quality assurance specialists. The development process is long, requiring millions of lines of program code that cost millions of dollars (Bethke, 2003). However, recent developments have provided a glimmer of hope for aspiring amateur authors. Some games, such as Neverwinter Nights (NWN) (http://nwn.bioware.com) and Oblivion (http://www.elderscrolls.com/games/oblivion_overview.htm) include story construction tools such as BioWare Corp.'s Aurora Toolset (http://nwn.bioware.com/builders) and Bethesda Softworks' The Elder

Scrolls (TES) Construction Set (http://cs.elderscrolls.com) respectively. Amateur authors use these tools to create game stories and post them on the Web. These community-authored stories are then played by hundreds of thousands of individuals. For example, the most popular of the 5,151 community-authored stories for NWN posted on the NWN Vault Web site (http://nwvault.ign.com) has been downloaded over 268,000 times as of July 2007, and the tenth-most, downloaded over 99,000 times. Unfortunately, the use of such story construction tools is beyond the capabilities of most individuals, including students, due to the computer programming skills required to use them.

There have been few game development studies involving children and adolescents. Kafai (1996) describes a game design research project that involved 16 fourth graders working over a period of six months to produce educational games that teach fractions. Students spent 92 hours using LogoWriter (http://el.media.mit.edu/Logo-foundation/logo/index.html) to create a game and spent another 20 hours on project-related activities in their school. Various forms of qualitative data were collected (e.g., observations, program code, student design plans, and student reflective notes). A major finding of this study was that the design process was more important than the actual game (artifact) produced by the students. According to Kafai (1996), "... students learned not only through design, but also about design, and reached a level of reflection that went beyond traditional school thinking and learning" (p. 94). In essence, the product was not as important as the learning process and the cognitive skill development that resulted from the activity (Fischer & Immordino-Yang, 2002).

Minkel (2002) describes a project where children created their own computer games. About 20 children in the inner city of Austin Texas (no ages provided) attended two four-day workshops learning about the development of computer games. The project was organized by software and computer games designer Allan Watts (creator of Noggimation). The children worked in teams to create the game rules, graphics, and sounds. They observed Watts using a laptop and a data projector as he programmed their contributions using Macromedia Director Shockwave Studio and ran their games inside of a Web browser. The complex coding was considered to be too difficult for the participants; educational gains were attributed to children being able to work in a team setting (Mitchell & Savill-Smith, 2004).

Recently, Robertson and Good (2005) held a four-day game creation workshop where ten students, aged 12 to 15 years, learned to build video game stories using the NWN computer game system. The purpose of the workshop was to enable students to write stories in the form of a computer game "and to develop narrative skills such as character creation, plot planning and interactive dialogue writing" (p. 57). The workshop involved a number of facilitators (e.g., a visual artist, a professional storyteller, an experienced amateur NWN game designer) and consisted of a number of activities, including group discussions about games, character design, model making, plot planning, storyboarding with digital cameras, and learning to author using the Aurora toolset. The toolset allows for the creation of 3D settings, characters, as well as the scripting of interactive plots and character/prop behaviors. A qualitative research study was conducted as part of the workshop experience. The most important finding was the high level of motivation expressed by the children to continue to work on game design and development. However, the participants had some difficulty using the Aurora toolset and felt that it somewhat constrained their ability to express story ideas. For example, participants had trouble using the Aurora plot wizard and various aspects of the Aurora toolset interface.

In summary, few projects have attempted to place young people in the role of game story authors without extensive support structures. The ability of young people to design and build computer games or game stories has been severely limited by the toolsets available. Consequently, their ability to create, express and construct ideas in a computer game medium is constrained. The notion of students building and forming multiple representations of knowledge, by creating sharable artifacts, is at the core of the constructionist approach to teaching and learning (Papert, 1991). The central tenant of constructionism is that learners are more likely to generate new ideas when they are actively engaged in developing some type of external artifact—a story, robot, or a computer game. Penner (2001) goes one step further, arguing for the importance of learners using computational tools to design and build interactive artifacts (models) that are open to examination, assessment, personal reflection, and public discussion. The building of a complex computer game or game story clearly represents such an artifact. In other words, if students are provided with powerful and viable toolsets, they should be able to construct interactive computer game stories.

3. Computer games and interactive stories

Viewing computer game adventures, especially computer role-playing game (CRPG) adventures, as stories is a contentious issue in the academic computer games literature. However, Murray (2005) makes a strong argument that we should not be distracted by such debates. It is outside the scope of this paper to engage in this debate, beyond an acknowledgement that there is indeed a controversy with respect to viewing games as stories. In this paper, we take the viewpoint that CRPGs are interactive stories and that constructing a CRPG is a form of story authoring.

Stories (narratives) are viewed by Bruner (1986) as one of two possible modes of thought, the other being the logico-scientific mode. Thus, story authoring is an essential aspect of human expression and communication of ideas. As McLellan (1994) points out, "stories are one of the most fundamental and powerful synthetic experiences available to us" (p. 76). Tompkins (1982) provides seven reasons why children should write stories. Story writing is a) an entertaining activity, b) fosters artistic expression, c) allows children to explore the functions and values of writing, d) stimulates imagination, e) helps them clarify and refine their thinking, f) supports their personal discovery and search for identity, and g) is an interesting way to learn to read and write.

In this paper, we use the term interactive story to refer to a computer role-playing adventure. Specifically, the story uses a set of cohesive computer-based content that is created by one or more authors. This content is presented to a single user (single player) or set of users (multi-player) by a computer game engine and user interface. We will focus on single user stories. A story instance is a single trace through the content that is experienced by the user. Each story is interactive since the user can directly affect the trace that is experienced. The story framework is the set of all possible traces through the content that could be experienced by a user.

The story content consists of a set of areas that comprise the story world. Each area consists of a set of tiles that form the physical landscape, a set of physical objects placed in the area, a set of creatures that inhabit the area, and the sounds and music that play when the player character (PC) travels through the story world. The story content also includes the interactive conversations that some creatures called non-player characters (NPCs) can have with the PC, the narration that can be displayed, and the specific physical interactions that can occur among the PC, the physical objects, and the creatures that inhabit the world. To foster re-use of story content between different interactive stories, pre-built component sets are usually available to authors. For example, story tiles are usually selected from pre-constructed tile-sets that have themes such as indoor building, indoor caves, outdoor city, outdoor country, cold mountainous, dessert, etc. The author can select creatures and physical objects, from pre-built libraries that contain textured models and pre-designed animations, and customize them to have specific characteristics, such as color and size that are appropriate to the context of the story. An author creates an interactive conversation graph for each NPC in the story that can converse with the PC. The conversation forms a graph since the player can select from a set of responses at various points in the conversation and each response can lead to a different set of responses for the NPC. Authors also write narration that can be displayed when different events occur in the story, and control the appearance of this narration using author-created scripts (small pieces of computer code). Authors also use scripts to create the interactions between the PC and the physical objects and creatures in the world.

4. Tools for interactive story authoring

Authors use sophisticated tools to create content for interactive stories. In this paper, we describe the set of tools that can be used to create interactive stories that can be played using the NWN game system. The basic tools are included in a single application called the Aurora Toolset created by BioWare Corp. that accompanies the game. Here are the production steps required to create an NWN game story:

1. Start the Aurora Toolset and create and name a new game story (module).
2. Create one or more areas by selecting terrain tiles from pre-built tile-sets.
3. Place pre-built physical objects (placeables, items, doors, creatures, triggers, waypoints, etc.) at specific locations in the areas.
4. Customize the placed objects using dialog boxes.

5. Create interactive conversations and attach them to particular NPCs.
6. Select sound effects and music from libraries and place them at particular locations in the areas so that when the PC comes near to these places, appropriate sounds are played.
7. Write scripts that allow the PC to interact with the objects in the game.
8. Save the module, exit the Aurora Toolset and launch the game story in NWN.

All of these steps can be performed using the Aurora Toolset. For example, the student-authored story called *MyShortStory122* contains two areas, *Exterior* and *Castle*. Fig. 1 shows a screenshot of the Aurora Toolset that displays a portion of the *Castle* area. There are two tiles in the foreground and four more tiles in the background. A placeable is an object that can be placed at any location in the game, but is too big to be carried by an NPC or the PC in their inventory. Trees, furniture, fountains and chests are examples of placeables. The author has put a placeable named "Gong of challenge" where two tiles meet and is customizing the placeable using a dialog box.

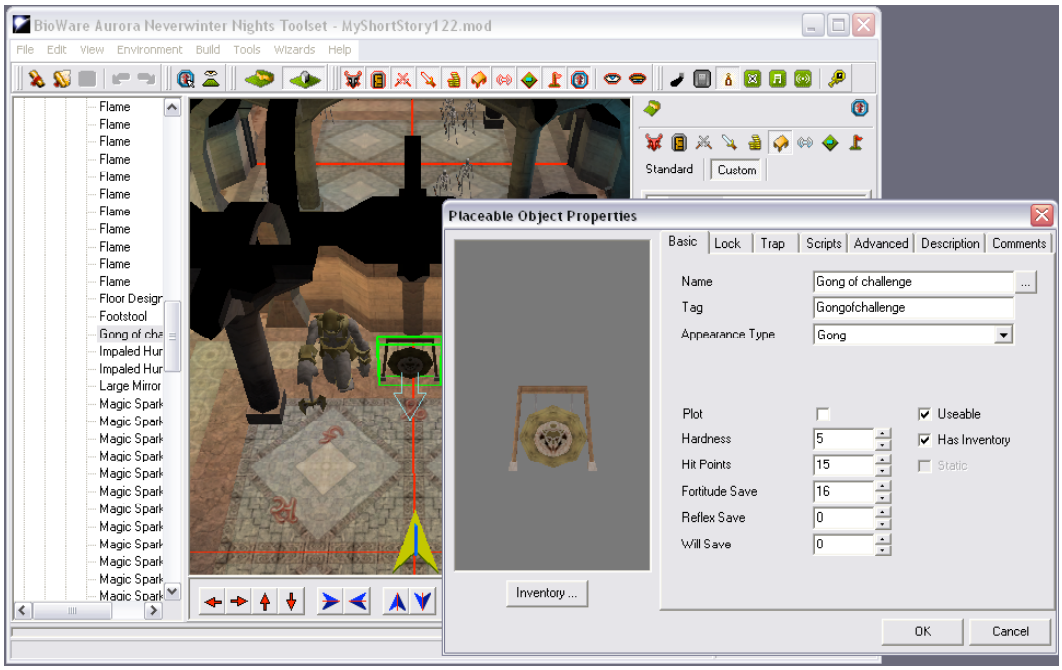

Fig. 1 Using the Aurora Toolset to place and customize a placeable object.

We created a tutorial that helps students learn how to use the basic components of the Aurora toolset to create their story worlds. Fig. 2 is an excerpt from a section of the tutorial (http://www.cs.ualberta.ca/~script/docs/TutorialFeb2007.doc) that explains how to create placeable objects. We include it to engender some understanding of the complexity of story creation tools and the level of explanation required. The tutorial contains figures similar to Fig. 1 that we omit here for brevity.

Besides placeable objects, the 26 page tutorial also covers items, doors, secret doors, creatures, factions, locked doors and keys, inventories, customizing objects, and conversations. In our experience, these are the toolset components that are most needed by student authors. We do not cover the creation of physical areas using tilesets. Instead, we provide students with a pre-built area to save time. Adding instruction for creating physical areas would require several additional pages and take students an additional one to two hours to learn.

Tutorial 2: Placeable Objects

Placeables include objects like chests, barrels, tables, trees, stones, signposts, etc. In this tutorial you will add some placeables to the castle to make it feel less empty.

1. If you have not already done so, start the Toolset and load the CastleYourName module.

2. Open the "Castle" area by double-clicking its name or selecting "View Area" from its context menu in the Module Contents panel.

The window on the right side of Aurora is called the "Palette", and contains a list of blueprints. Blueprints are designs, not actual things. You use a blueprint by "painting" an object in the Area window.

3. Open the Placeable palette by pressing the table icon located in the game objects palette. Ensure the Standard button is selected.

4. Left-click on "Table" in the "Miscellaneous Interior" category.

5. Paint the table into the castle by left-clicking where you want it to go. Put it in the kitchen, which is the area labeled 1 on your map.

6. Paint a chair near the table. You can find "Chair" blueprints under the "Miscellaneous Interior" category. Pick the first one.

7. If you want the chair to face in towards the table, you can turn it. Make sure your new chair is highlighted with a green box by clicking on it. Then use the two rotation buttons to rotate the chair. The blue arrow shows which way the chair is facing.

8. Save your module.

Fig. 2 An excerpt from the Aurora Toolset / ScriptEase tutorial on placeables.

All of the production steps except for step 7 (write scripts) are easy to perform using the Aurora Toolset, as evidenced by the success of the pilot and case studies described in the next two sections. The activities in each of the other steps involve selecting menu items, clicking a mouse button on locations, using checkboxes or radio buttons, and typing simple text, such as the names of objects and dialogue to be spoken. However, performing step 7 using the Aurora Toolset is impossible for most students. To create scripts, the author must write statements in a C-like language called NWScript and attach them to objects, NPCs, and specific points in conversations. These scripts are essential to make something happen in the game when a PC interacts with an object. For example, an author could place a script on a placeable object ("Gong of challenge") so that when the PC places a specific item ("Thors hammer") into its inventory, a creature would be spawned. Fig. 3 shows how the Aurora Toolset is used to write this script in the NWScript language. In this case, the author would have to literally write a program that calls an API (Application Programmer Interface) provided by BioWare. For example, the second line of the script shown in Fig. 3 calls an NWScript function called `GetInventoryDisturbItem()`, to find out which item the PC placed into the "Gong of challenge". The fourth line is an "If" statement that checks to ensure that the action done by the PC was to add an item to the "Gong of challenge" rather than remove an item and that the item added was actually "Thors hammer".

As a second example, an author could place a script on a conversation point so that when that point is reached in a conversation ("Take this key"), a key is transferred from the NPC's inventory to the inventory of the PC. However, the author may only want the "Take this key" conversation option to appear after the PC has completed a quest. In this case, the author must write another script that controls whether this statement appears in the conversation. In summary, if a game story author wants to use the Aurora Toolset to provide any meaningful interactions between the PC and game objects, then the author must be an experienced programmer.

Writing scripts is a very difficult part of interactive story authoring and the problems have been well documented (McNaughton, Cutumisu, Szafron, Schaeffer, Redford, & Parker, 2004). The ScriptEase tool has been created to allow authors to generate scripts without writing any NWScript code. Instead, the author selects from frequently occurring story patterns, adapts the pattern to the context of the story being authored, and then pushes a button so that the pattern generates the script code. An author can use ScriptEase to replace production steps 7 and 8 by the following steps[1]:

7.1 Save the module and exit the Aurora Toolset.
7.2 Start ScriptEase and open the module.
7.3 For each desired interaction select an object and create a pattern instance.
7.4 Adapt each pattern instance for the story.

[1] Note that steps 7.1, 7.2 and 7.6 would disappear if ScriptEase was incorporated directly into the Aurora Toolset.

7.5 Select a menu item to generate the scripts and save the module.
7.6 Exit ScriptEase and launch the game story in NWN.

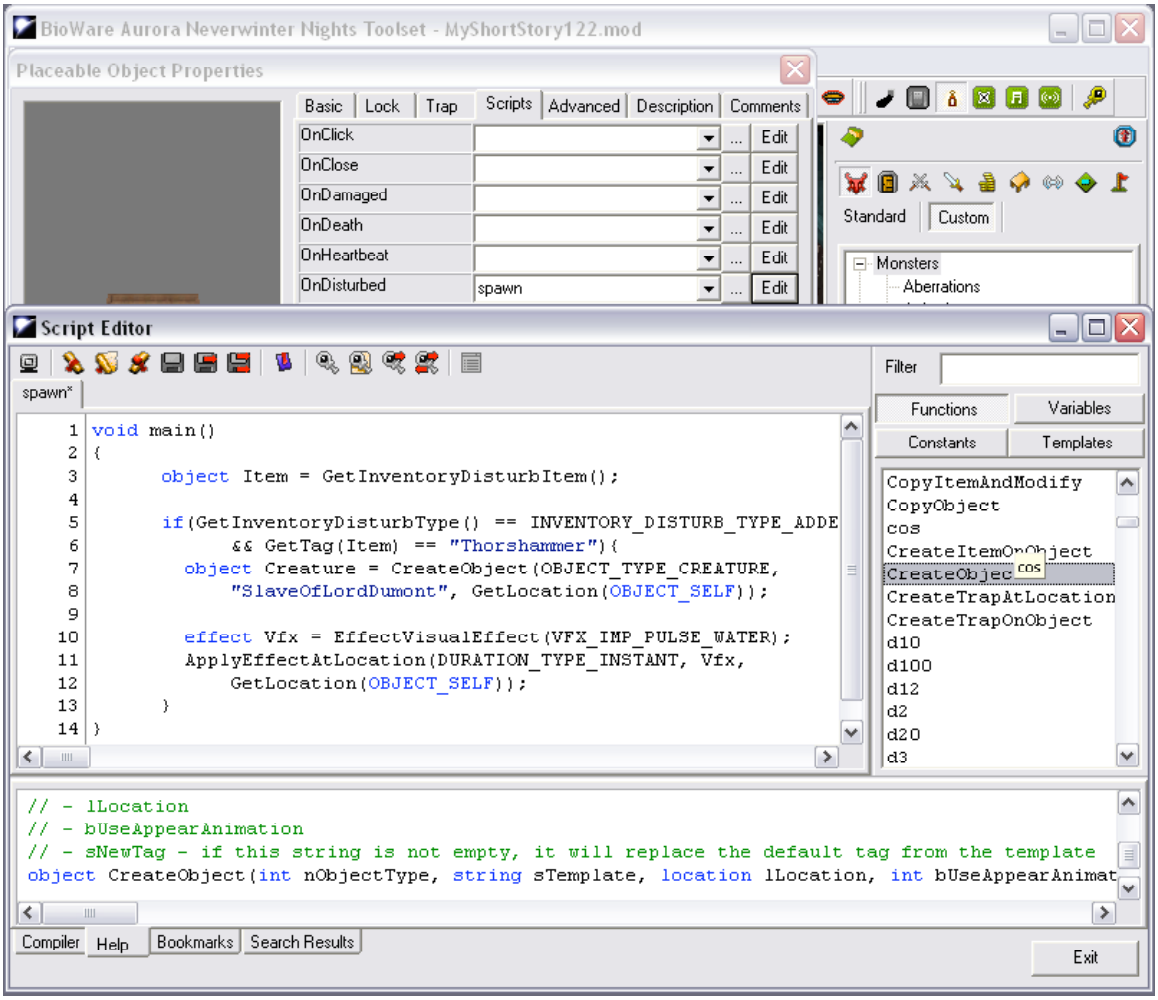

Fig. 3 Using the Aurora Toolset to script a placeable object.

ScriptEase reads game stories that are saved by the Aurora Toolset and simplifies the process of creating, editing and testing scripts. For example, rather than writing the NWScript code shown in Fig. 3, a story author could open the game story module in ScriptEase and create an instance of the *Placeable add – (specific item) spawn creature* pattern by selecting its pattern name from a menu and applying it to the "Gong of challenge" object. This pattern would be adapted by setting four options using menus, the placeable ("Gong of challenge"), the specific item ("Thors hammer"), the appropriate kind of creature to spawn ("Slave of Lord Dumont"), and the spawn effect ("Pulse, Water"). After selecting the "Compile and Save" item from a menu to generate the NWScript code automatically, the author could exit ScriptEase and launch the game story. When the PC places "Thors hammer" in the "Gong of challenge", the generated script would trigger and the "Slave of Lord Dumont" would spawn with a "Pulse, Water" visual effect.

Fig. 4 shows a snapshot of the ScriptEase adaptation step. The placeable container ("Gong of challenge") is being selected from a menu that looks identical to the menu in the Aurora Toolset that was used to create the "Gong of challenge". Note that the pattern in Fig. 4 has been opened to show its natural language description, but in fact the author did not have to open the pattern to adapt it by setting the options (*The Placeable, Specific Item, Creature Blueprint* and *Spawn Effect*). Our studies (Szafron, Carbonaro, Cutumisu, Gillis, McNaughton, Onuczko, et al., 2005) show that authors who have no programming skills can use ScriptEase to automatically generate NWScript code for them.

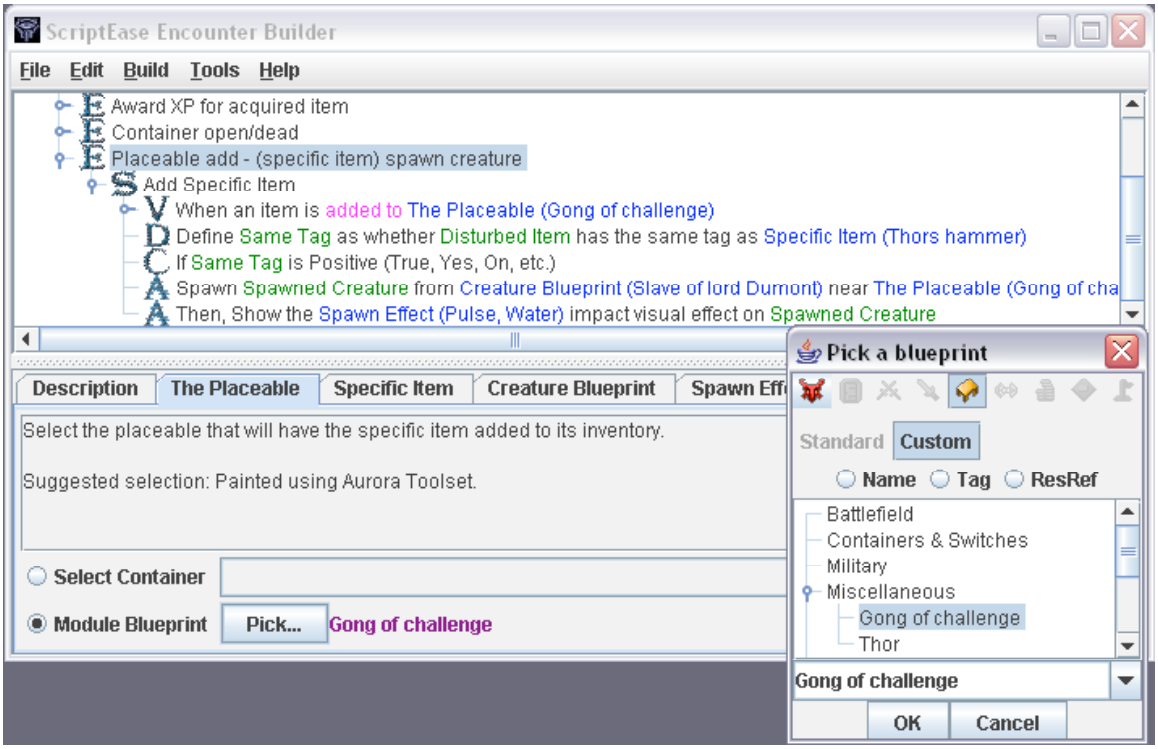

Fig. 4 Using ScriptEase to adapt a pattern to a story by setting options.

The kind of adaptation described here is the simplest one (setting pattern options). However, the author can make more sophisticated adaptations. The tutorial we referenced earlier does not provide instructions on how to manually write scripts in the Aurora Toolset. Instead, it provides instructions on how to use ScriptEase to generate scripts. We include another excerpt from this tutorial as Fig. 5 that describes how to use ScriptEase to add an action to a pattern – a slightly more sophisticated adaptation than setting pattern options. In this example, we use the pattern *Placeable remove (specific item)*. The author must adapt the pattern by adding the specific action that should happen when the specific item is removed from the placeable. Again, this excerpt is provided so that the reader can gain some understanding of the complexity of story creation tools and the level of explanation required. The tutorial contains figures similar to Fig. 4 that we omit here for brevity.

Tutorial 6: Actions

Now that we have the book in the statue's inventory, we can give the player a message when she takes it.

1. Close your module in the Toolset and open it in ScriptEase.
2. Right-click on the CastleYourName.mod folder and pick "New Specific Encounter" from the menu. Select the "Book Statue" in the left hand list and "Placeable remove – (specific item)" from the right hand list.
3. Set the Specific Item tab to "Origin of Magic".
4. Expand the "Placeable remove – (specific item)" encounter.

The encounters we create in ScriptEase contain "situations", represented by the symbol S. Situations tell the game how to respond when the player reaches a point in the story. Sometimes an encounter will have situations that you don't need. These are there because it's easier to delete things you don't need rather than create everything from scratch.

A situation contains actions, represented by the symbol A, that the game does when the player reaches that part of the story. We want an action to make a message appear when the player removes the "Origin of Magic", telling her to put it in the other statue.

5. Right-click on the "Remove Specific Item" situation and pick "Add an Action → Action Atom → Text → Make an object speak text" from the menu.

6. The situation will expand, showing the new action.

7. On the Speaker tab, click the "Select Object" option, and then pick "The Placeable" from the drop-down list beside it.

Phrases in ScriptEase like "The Placeable" and "The Item" are variables, like the "x" and "y" you use in formulas in math class. In ScriptEase, the Encounters are formulas for how the story progresses. Just like in math, you can use the same formula with different values for the variables.

8. On the Text tab, in the "Constant" box, type "Place the book in the other statue to help find your Grandmother." DO NOT include the quotes.

9. Leave the Volume tab on the "Constant" option set to "speak".

10. Save and Compile, then try out the changed game!

Fig. 5 An excerpt from the Aurora Toolset / ScriptEase tutorial on ScriptEase adapting patterns by adding actions.

The only technology preventing amateur authors from authoring interactive stories is a tool to aid in scripting interactions – manual scripting is too hard. Our solution is ScriptEase, a tool that allows non-programmer to use generative patterns to create program scripts automatically. More detailed descriptions of ScriptEase include its user interface (Cutumisu, Onuczko, McNaughton, Roy, Schaeffer, Schumacher, et al., 2006) and pattern catalog (Onuczko, Cutumisu, Szafron, Schaeffer, McNaughton, Roy, et al., 2005). The question asked and answered in this paper is: can ScriptEase be used effectively in the classroom?

5. Interactive story authoring in the high school English curriculum

We conducted a pilot and then two case studies in which grade ten high school English classes used the Aurora Toolset and ScriptEase to author interactive stories in the NWN game world. In each case, high school students wrote a traditional pen and paper short story (TS) and authored an interactive computer game story (IS). The pilot was used strictly to obtain feedback on the instructional materials, to improve them and to ensure that the tools worked well. For example, feedback from this pilot led to two of the original tutorials, one for the Aurora Toolset and one for ScriptEase, being extensively revised and combined into a single tutorial (Szafron et al., 2005). No data was gathered during the pilot.

We had three overall goals for the case studies. First, we wanted to determine whether high school students were capable of using tools to author interactive stories. Second, we wanted to determine whether any group of students would be disadvantaged in their ability to write interactive stories using the computer game development medium, due to 1) previous computer programming experience, 2) gender, 3) the amount of prior computer game playing experience, 4) the amount of prior experience using the Web for surfing and chatting, 5) creative ability, or 6) general intellectual ability. Third, we wanted to determine whether some students who do poorly relative to their peers on traditional writing exercises would exhibit relative improvement in interactive story authoring.

Each case study was conducted with a single grade ten high school English class. Data from both classes was merged to obtain enough data for analysis. Each class was from a different high school, but both schools are located in urban middle class settings in the same Canadian city (Edmonton). Edmonton is a multi-ethnic city and this was reflected in the case study population where 13 out of the 37 participants were fluent in another language besides English. These languages were Albanian, Arabic, Cantonese, Farsi, French, Hindi, Mandarin, Polish, Punjabi, Russian, and Spanish, although some students were fluent in more than one of these languages.

Each of the high school classes received instructions on how to write a short story, as part of the regular high school English curriculum. Each student then wrote a short story that was scored using the rubric in Table 1, based on the standard elements of a short story: Characters, Setting, Plot, Conflict, and Theme. The same high school teacher scored all of the traditional stories from both classes. The total number of students was 37 (21 females and 16 males).

Each class was introduced to the concept of interactive stories and given the assignment to author an interactive short story that could be "played" in NWN. In each case study, the high school student authors took part in a two-day workshop conducted at the University of Alberta. The workshop consisted of two tutorials (total time six hours), along with limited time (two hours) to start their interactive short story. The

first tutorial (http://www.cs.ualberta.ca/~script/supplementary01.html) showed them how to play an NWN game story. The second tutorial http://www.cs.ualberta.ca/~script/supplementary01.html) showed them how to construct an interactive story for NWN using both the Aurora Toolset and ScriptEase.
The teacher and some ScriptEase researchers supervised the students and answered questions. During the day, there were two extended breaks where the students were exposed to research projects that were of interest to this age group. At the end of the workshop, the students returned to their high school classrooms to spend four more hours to complete their stories.

Table 1
The common rubric used to grade both the traditional short stories (TS) and the interactive stories (IS), where Style refers to grammar, style and organization. Note that interactive stories contained dialogue.

	Story effect achieved	Score
Characters	There is little character development. The characters are shells with little depth. Characters have little influence on theme. Character dialog is present, but simple and usually irrelevant.	2 or
	Few characters are developed adequately. The characters are somewhat revealed but reader/player has to make assumptions. The characters develop very little. Change in character is barely related to theme. Character dialog is present, but not very meaningful.	4 or
	The characters are developed adequately. The characters are revealed through a few methods of characterization. The characters are similar to the way they are introduced. Change in character is somewhat tied to theme. Character dialog is present.	6 or
	The characters are developed fully. The characters are revealed through multiple methods of characterization. The characters remain true to the way they are introduced or if they change, the change is logical and in keeping with the theme. The characters are involved in dialog with the dialog properly punctuated.	8
Setting	The setting is not connected to the plot or characters. There is little to no description of the setting.	1 or
	The setting is difficult to connect to the plot and characters in the story. The setting is very simplistic. Setting is simply stated.	2 or
	The setting somewhat fits the plot and characters in the story. The setting has a few details. Setting is developed using one of explicit or implicit details.	3 or
	The setting closely fits the plot and characters in the story. The setting has sufficient details for the reader to visualize the local. The setting is developed using both explicit and implicit details.	4
Plot	The plot is unclear. A plot conclusion is absent. It is difficult to distinguish plot elements.	2 or
	The plot is barely developed. It comes to a conclusion but there are some loose ends. Only a few plot elements are present.	4 or
	The plot is somewhat developed. Plot comes to a conclusion but it is not logical. Most elements of plot are present.	6 or
	The plot is developed fully. The plot comes to a logical conclusion. All 4 elements of plot are clearly present: Exposition, Rising Action, Climax, and Falling Action.	8
Conflict	It is difficult to distinguish the main conflict.	1 or
	One main conflict is present in the story. The conflict is partially developed.	2 or
	One or more conflicts are present in the story. Conflicts are somewhat developed and interesting. Some conflicts are resolved.	3 or
	One or more conflicts are present in the story. Conflicts are fully developed & interesting. Conflicts are resolved in understandable ways.	4
Theme	The story is missing a central idea or belief. It is difficult to understand the purpose of the story.	2 or
	Central idea or belief is barely developed and understood. Some use of voice, tone, characters, setting and mood are present. The theme is unclear.	4 or
	Central idea or belief is somewhat developed and understood through the use of voice, tone, characters, setting and mood. The theme can be deduced.	6 or
	Central idea or belief is fully developed and understood through the use of voice, tone, characters, setting and mood. The theme is meaningful and clear.	8
Style	Incomplete organization and poor use of syntax, weak style serious grammatical flaws.	1 or
	Basic vocabulary and style, some major grammatical flaws.	2 or
	Good word usage and vocabulary style is solid, a few grammatical flaws.	3 or
	Sophisticated and fluent style with a cohesive style and vocabulary no grammatical problems.	4
	Total Max 36	

The high school teacher provided the students with a module, *MyShortStory*, which constituted the basis for their interactive story. The module contained the interior and exterior of a medieval castle, without any objects (placeables, items, etc.) or scripting. The students were asked to author their interactive story using the given module by adding their own props and characters and generating appropriate scripts using ScriptEase. The students were provided with blank maps of the castle and they were required to annotate

the maps by hand according to their story, so that their stories were easier for the evaluator to navigate. Students were given three class periods of computer lab access to author their stories. However they did have additional access to the labs if they wished to work on their stories outside of class, but no assistance was provided. This allocation of three periods in the lab corresponded to the amount of time assigned to an equivalent traditional writing project. In all, the high school authors spent an average of four hours working on their stories in the high schools, plus two hours at the University, for a total of six hours. At the end of the allotted time, the stories were submitted and scored by the same teacher that scored the traditional story, using the same rubric. There is a case to be made to evaluate the interactive stories using a different rubric, since the expressive medium is different. However, for these studies we decided to use the same rubric (applied by the same high school teacher) to reduce the number of variables in the experiment and to make the comparisons much more straightforward.

In addition to performing the instructional tasks, each student completed an entrance survey (http://www.cs.ualberta.ca/~script/supplementary01.html), an exit survey (http://www.cs.ualberta.ca/~script/supplementary01.html), the Torrance Tests of Creative Thinking-Figural (Cropley, 2000; Kim, Cramond, & Bandalos, 2006; Torrance, 1999), and the Shipley Institute of Living Scale (SILS) as measure of general intellectual functioning (Zachary, 2000). The SILS has been shown to have a correlation of between .74 between .85 with the WAIS-R (Zachary, Crumpton, & Spiegel, 1985).

6. Results and discussion

Each sub-section provides an answer to one of three goals set for this study: a) can high school students successfully write interactive stories using the ScriptEase toolset, b) how do any of six factors we identified impact student performance, and c) are there students who perform poorly relative to their peers at traditional story writing, whose relative performance improves for interactive story authoring? When appropriate, we used Effect Size (ES) to evaluate the level of practical significance (Hojat & Xu, 2004). ES results are classified into four categories: none (less than 0.20), small (between 0.20 and 0.50), medium (between 0.50 and 0.80) and large (greater than 0.80) (Cohen, 1988).

6.1 Can high school students use the Aurora Toolset and ScriptEase to write interactive stories?

The first goal of our study was to determine whether high school students were capable of using our ScriptEase toolset to author interactive stories. The answer depends on what metric is used to define success. We identified five measures of potential success: Interactive Story (IS) scores, Information Technology (IT) scores, the number of pattern instances used in a story, the number of adaptations used in a story and answers to exit questionnaires. The exit questionnaires were also used as a mechanism to encourage the students to reflect on their educational experience as the capstone of the constructionist learning paradigm (Vygotsky, 1962).

6.1.1 Interactive story score

The rubric shown in Table 1 was used for assessing both the traditional stories (TS) and the interactive stories (IS). This rubric does not directly assess the use of ScriptEase patterns in an IS. Therefore, it is possible to use the Aurora Toolset, without ScriptEase, to construct an IS that contains no scripts. However, such a story would exhibit minimal interaction between the player character and the objects in the game, since such interactions must use scripts either generated by ScriptEase or manually-written using the Aurora Toolset. This lack of interaction could result in a boring and repetitious story, making it more difficult to score marks in each of the categories in the rubric. Therefore, the IS score directly records student success of using interactive stories in the classroom, and indirectly measures the success of ScriptEase.

The mean of IS scores is 21.76 (SD = 7.05) compared to the mean of 25.48 (SD = 4.40) for TS scores. The correlation between IS scores and the TS scores is $r = 0.35$. A t-test reveals a significant difference in the means ($t = 2.10$, df = 35, $p < .05$) with a corresponding ES of -0.63 that is considered medium. Therefore, students in the study scored significantly lower for their interactive story than for their traditional story. In fact, 10 out of 37 students scored less than 50% (the passing score) on their interactive stories while no students scored less than 50% on their traditional stories. These results should not be

surprising, since the students only had six hours of instruction on authoring interactive stories and they have spent many previous years in school writing traditional short stories. Based on the IS score, the answer to the question, "Are high school students capable of using our tools to author interactive stories?", is that with the allotted six hours of training time, 73% of the students in our study could score a passing grade (at least 50%) on their interactive story. Based on interactive story scores, the answer is that 73% of the students succeeded at authoring an interactive story.

6.1.2 Information technology score and number of patterns used

Each student in the study was also assigned an information technology (IT) score for their story. In addition to being part of the English curriculum, this activity was considered a valid experiential learning activity in the schools' Career and Technology Studies program. Table 2 shows the rubric for IT scores, based on the number of patterns used and the number that were different from the patterns they used during training.

The IT score (mean = 6.42, SD = 2.68) is a direct measure of the success of using ScriptEase to create scripts in interactive stories. The IT scores correlates, $r = 0.45$, with the IS score, even though that IS score was based on the rubric in Table 1 and does not include any measure of scripting. The correlation ($r = 0.17$) between the IT scores and the TS scores was not strong. There were only six students whose IT score was below 5, yielding an 84% success rate of students scoring more than 50%. Two students had an IT score of 0, indicating that they did not successfully create a single pattern instance in their story (one student tried to use an instance that did not work). As a direct measure of a student's ability to use our tools to create scripts in interactive stories, the IT score indicates a success rate of 95% of students were able to use at least one pattern.

Table 2
The rubric used to assign information technology (IT) scores to the interactive stories.

Patern use	Score
No patterns used	0
1-2 patterns instantiated, pattern instances don't actually do anything in the story	2
1-2 patterns instantiated, pattern instances do something in the game	4
Several patterns instantiated, all patterns used are from the tutorials	6
Several patterns instantiated, some patterns deviate from those found in the tutorial, most do not	8
Several patterns instantiated, most patterns deviate from what is found in the tutorial	10

We also counted the number of pattern instances (NP) used in each story. This number ranged from 0 to 15 (mean = 6.33, SD = 3.84) and correlated $r = 0.27$ with the IS score. The NP score was highly correlated ($r = 0.8$) with the IT score, thus no new insights can be gleaned from this relationship. The variation between the NP score and the IT score is due to the preference in the IT score for patterns that are different from the ones used in the training. Based on IT score, the answer is that 84% or 95% of the students were successful.

6.1.3 Number of adaptations used in the story

We counted the number of adaptations each student made to the pattern instances they created. ScriptEase supports a broad range of pattern adaptations that an author can make. The simplest kind of adaptation is setting pattern options, as illustrated in Fig. 4. However, the author can also add and remove components from the patterns (Cutumisu, Onuczko, et al., 2006). For example, the pattern shown in Fig. 4 has two actions: a creature is spawned and a visual effect is fired. These actions are included in the pattern when it is instantiated. However, the author could use a menu to add a third custom action, such as playing a "gong sound". In general, the more adaptations an author makes, the more unique the interactions between the player and the environment. The number of adaptations (excluding option setting) used by the student authors in their stories is another measure of this ability to use ScriptEase to create interactive stories. The number of adaptations (excluding option setting) ranged from 0 to 36 (mean = 11.57, SD = 10.57) and the correlation with IS score is $r = 0.39$. Only three students did not make any adaptations to their story, other than setting options. This indicates that 92% of the students were able to perform pattern

adaptations during their interactive story authoring. It could be argued that to write the most expressive (non-repetitive) stories, pattern adaptation is necessary. By this measure, 92% of the students succeeded.

6.1.4 Reflection - exit questionnaires

What do students think about the difficulty of writing interactive stories? To encourage students to reflect about interactive story authoring, we asked them to complete an exit survey (http://www.cs.ualberta.ca/~script/supplementary01.html), after both the interactive and traditional stories were complete. Table 3 shows the results of student introspection about the difficulty of authoring an interactive story.

Table 3
Student reflection about the difficulty of writing an interactive story. Each factor listed in this table began with the preface "How difficult was it to…". The choices were "easy" – 0, "a bit hard" – 1, "fairly hard" – 2, "very hard" – 3 and "way too hard" – 4. Note that 3 students of the 37 did not complete the exit questionnaire.

Factor	Number 0's	Number 1's	Number 2's	Number 3's	Number 4's	Mean	Standard Deviation
"Read" the NWN tutorial story (play the tutorial game story)	14	18	2	0	0	0.65	0.60
Learn to use the Autora Toolset	19	11	4	0	0	0.56	0.70
Learn to use ScriptEase	7	9	16	1	1	1.41	0.96
Understand the terminology of interactive stories	11	15	7	1	0	0.94	0.81
Write your own interactive story	8	15	6	3	2	1.29	1.12

The "difficulty" assigned by the students matches the researchers' expected ranking of the difficulty of the activities. Reading an NWN story (playing a sample game) and learning to use the Aurora Toolset are easiest (0.65 and 0.56 respectively). One might expect that playing the game should be easier than using the Aurora Toolset to create game objects. However, it does take some concentration and practice to learn how to control the PC in the story. That is why there are more 1's for learning to play the game than for learning to use the Aurora Toolset. Learning to use ScriptEase is harder, since it involves more abstraction skills to understand and pick patterns and more logical thinking to adapt the patterns. It is interesting that students found it relatively easier to write an interactive story (mean difficulty 1.29) than it was to learn how to use ScriptEase (mean difficulty 1.41). If we use perceived difficulty as a measure of success, then 85% (29/34) of the students found the activity below the threshold of very hard.

6.1.5 Summary of success

In summary, 95% of the students were able to use ScriptEase patterns to generate scripts for an interactive story and 92% were able to adapt the patterns they used beyond simply setting options. We found that 84% percent of the students were able to use ScriptEase to generate a substantial number and variety of scripts (IT score of at least 50%). The success rate for using both the Aurora Toolset and ScriptEase to generate an interactive story that achieves a passing IS score (at least 50%) based on the rubric of Table 1 was 73%. Note that being able to use ScriptEase to generate a substantial number and variety of scripts did not guarantee a passing grade in the IS score, since the IS score is based on a traditional story evaluation rubric – good scripts do not necessarily mean good characters, plot, setting, etc. There was a significant difference between the means of the IS score and TS score indicating a medium Effect Size in favor of success at writing a traditional story. It would be interesting to discover whether increasing the training time from six hours would have a measurable positive effect on the IS success rate. However, conducting such a study is problematic, since finding the extra time required may not be possible within the current high school curriculum requirements. Finally, based on student reflection, 85% of the students found the activity to be "easy", "a bit hard" or "fairly hard" as opposed to "very hard" or "way too

hard". By any of the measures we have presented, most students were successful in using the Aurora Toolset and ScriptEase to author an interactive story.

6.2 Are students disadvantaged by any of six identified factors?

The second goal was to determine whether any of six factors would influence a student's ability to write an interactive story: a) previous computer programming experience, b) gender, c) the amount of prior computer game playing experience, d) the amount of prior experience using the Web for surfing and chatting, e) creative ability, or f) general intellectual ability. To measure each of the first two factors, the students were split into two groups based on their previous programming experience and their gender. The IS scores of the two groups for the first two factors were compared using a *t*-test and ES. The results are shown in Table 4. Each of the other four factors was coded as an integer value in a fixed range and a correlation was computed between the integer value of the factor and the IS score. The results are shown in Table 5.

Table 4
Programming experience and gender, relative to IS score.

Factor	Group	Number	Mean Story Score	Standard Deviation	t-value	Significance at $p < .05$	Effect size value	Effect size
Programming experience	yes	12	21.92	7.54	0.59	0.56	-0.21	small
	no	25	23.40	6.91				
Gender	female	21	22.95	7.24	0.03	0.97	+0.01	none
	male	16	22.88	7.03				

Table 5
Correlations of four measured study factors with IS score.

Factor	Mean	Standard Deviation	Correlation
Amount gaming	7.24	11.32	0.13
Amount Web chatting/surfing	14.38	11.48	0.12
Creative ability	105.62	12.76	0.09
Intellectual ability	99.14	7.09	0.25

6.2.1 Prior programming experience

To determine whether prior programming experience would affect a student's ability to succeed at interactive story authoring, we included a question in the entrance survey about computer programming experience. Overall 32% of the students indicated they had programming experience, 14% of the females (3/21) and 56% of the males (9/16). The mean IS score of those with prior programming experience was 21.92 (SD = 7.54) while the mean IS score of those with no previous programming experience was 23.40 (SD = 6.91). There was no significant difference between the two groups ($t = 0.59$, $df = 35$, $p < .05$) and the ES was calculated at -0.21, which is considered to be small. Therefore, prior programming experience does not appear to be a significant factor that affects interactive story authoring and may even be a slight hindrance (see Table 4).

6.2.2 Gender differences

We were also interested in determining whether males (43%) or females (57%) would perform differently on the interactive story authoring task, given that the males had significantly more previous programming experience and significantly more time playing computer games (see Table 6). However, the results reported in Table 4 indicate no difference between the IS scores of the females versus males ($p < 0.05$). The ES of 0.01 is considered none. Therefore, gender does not appear to be a factor that correlates with student performance scores for interactive story authoring.

Table 6
Study measures factored by gender and sorted by effect size

Measure	Gender	Score	Standard Deviation	t-value	Effect size value	Effect size
Game Hours	female	2.63	2.43	4.03 *	-1.24	large
	male	15.31	14.23			
Programming Experience	female	0.14	0.36	2.93*	-0.95	large
	male	0.56	0.51			
Creativity	female	112.52	10.80	2.36*	+0.77	medium
	male	103.13	13.48			
Surf/Chat Hours	female	18.24	12.01	1.81	+0.61	medium
	male	103.13	13.48			
Traditional Story (TS) score	female	25.19	3.88	0.85	-0.28	small
	male	26.44	5.05			
Information Tech. (IT) score	female	59.76	23.74	0.29	-0.09	none
	male	62.38	31.00			
Intellectual ability	female	103.95	6.11	0.03	+0.01	none
	male	103.88	8.42			
Interactive Story (IS) score	female	22.95	7.24	0.03	+0.01	none
	male	22.88	7.03			

* $p < .05$

Although gender did not affect interactive story authoring success, we did find some interesting gender differences during this study (see Table 6). Significant differences were found on measures of a) time spent playing computer games, indicating that males who participated in the study spent more time playing computer games than the females, b) programming experience, indicating that males who participated in this study had more programming experience than the females, and c) creativity, indicating that females who participated in the study were more creative than the males.

There were no statistically significantly differences on time spent surfing and chatting, but the results did indicate that females are more likely to spend time on these Web activities than the males (ES = 0.61, medium). Furthermore, the Effect Size differences on IT, IS, and intellectual ability were none, with the mean scores on both intellectual ability and IS scores showing only a .07 difference. It is interesting that, although male students had significantly more prior programming experience, there was no difference between the females and males in terms of their ability to use and adapt patterns as measured by the IT score. This directly supports our contention that ScriptEase can be used by non-programmers.

6.2.3 The amount of prior computer game play experience

To determine whether the amount of prior computer game playing experience would affect a student's ability to succeed at interactive story authoring, we included questions in the entrance survey about the number of weekly hours spent playing each of three different kinds of computer games: console games (e.g., X-box, Gameboy, Playstation, etc.), computer games (e.g., Doom, Sims, Starcraft, etc.), and Internet games (e.g., Ezone, Pogo, etc.). We added the answers for each student to obtain a total weekly game playing amount. The correlation of total weekly game play hours versus IS score is $r = 0.13$. Therefore, the amount of prior computer-game play experience does not appear to be a factor that affects a student's ability to write good interactive stories. The results are presented in Table 5.

6.2.4 The amount of prior web chatting and surfing experience

To determine whether the amount of prior experience with Web chatting and surfing would affect a student's ability to succeed at interactive story authoring, we included questions in the entrance survey about the weekly number of hours spent using the Internet for chatting and surfing. We added the answers for each student to obtain a total weekly amount. The correlation between students' total weekly amount

and their IS score was only $r = 0.12$. Therefore, the amount of prior Web chatting and surfing experience does not appear to be a factor that affects a student's ability to write good interactive stories (see Table 5).

6.2.5 Creative ability

To estimate creative ability, study participants were tested using the Torrance Tests of Creative Thinking-Figural (Cropley, 2000; Kim, et al., 2006; Torrance, 1999). We then tried to determine whether creativity (as measured by this test) would affect a student's ability to succeed at authoring an interactive story using our tools. The correlation of creativity versus IS score is only 0.09. Therefore, the creativity as measured by this test does not appear to be a factor that affects a student's ability to write good interactive stories (see Table 5).

6.2.6 Intellectual ability

Study participants were tested using the Shipley Institute of Living Scale (SILS) test of intellectual ability (Zachary, 2000). The SILS is a group test that can be administered in a short time frame (less than 30 minutes) and has been shown to have a correlation between $r = 0.74$ and $r = .85$ with the WAIS-R (Zachary et al., 1985). We then tried to determine whether intellectual ability (as measured by this test) would affect a student's ability to succeed at authoring an interactive story using our tools. The correlation of SLIS score versus IS score is only $r = 0.25$. Therefore, the intellectual ability as measured by this test does not appear to be a large factor that affects a student's ability to write good interactive stories (see Table 5). It is interesting that the correlation of SLIS score to the TS score is higher at $r = 0.37$, but still not significant.

6.3 Can some students shine using an alternative story authoring medium?

The third goal was to determine whether some students who do poorly relative to their peers on traditional writing exercises would exhibit relative improvement in interactive story authoring. If this is the case, interactive story authoring may provide these students with an expressive medium that would allow them to exhibit their talent in a way not available to them through traditional writing.

There was a group of students (21/37) who scored relatively better on their interactive story than on their traditional story. However, this ratio by itself is insignificant, since one would expect about half the students to do relatively better on each exercise and 21/37 is close to half. However, the correlation between IS score and TS score was only $r = 0.35$ (small ES), indicating that there is a reasonable difference in the rankings of the students for the two activities. Therefore, interactive story authoring is a good alternate activity for providing pluralistic success in the classroom.

We next tried to identify any single attribute or combination of attributes (from those we measured) that could serve as a significant predictor of which students would benefit most from interactive story authoring over traditional story writing. We failed to find a significant combination of attributes that could be used as a predictor. However, we did notice one weak gender-based correlation that would require a much larger study to confirm or refute. We computed the correlation between each authoring mode (traditional and interactive) and the SLIS score, and between each authoring mode and the creativity score – both for all test subjects and by gender. We then computed the Fisher Z-transform differences of these correlations to determine the Effect Sizes of the differences (Hojat & Xu, 2004). The results of the SLIS scores are shown in Table 7 (the effect sizes for the creativity tests were all none so they are not shown).

This table provides very weak evidence that female students with low SLIS scores could benefit the most from interactive story authoring as an exercise to provide an alternate expressive medium. Fig. 6 shows a plot of this data. The five students who raised their scores the most are represented by unique icons. Those female students who improved their relative performance on the interactive activity relative to the traditional activity by the largest amounts have some of the lowest SLIS test scores. Interactive story authoring provided these students with an expressive medium that allowed their talents to shine.

Table 7
Comparisons of correlations between authoring scores and SLIS scores

Test and group	Authoring Mode	r value	Effect size value of Z-transformed difference	Effect size
SLIS all	traditional	0.37	+0.13	none
	interactive	0.25		
SLIS males	traditional	0.33	-0.03	none
	interactive	0.36		
SLIS females	traditional	0.45	+0.35	small
	interactive	0.14		

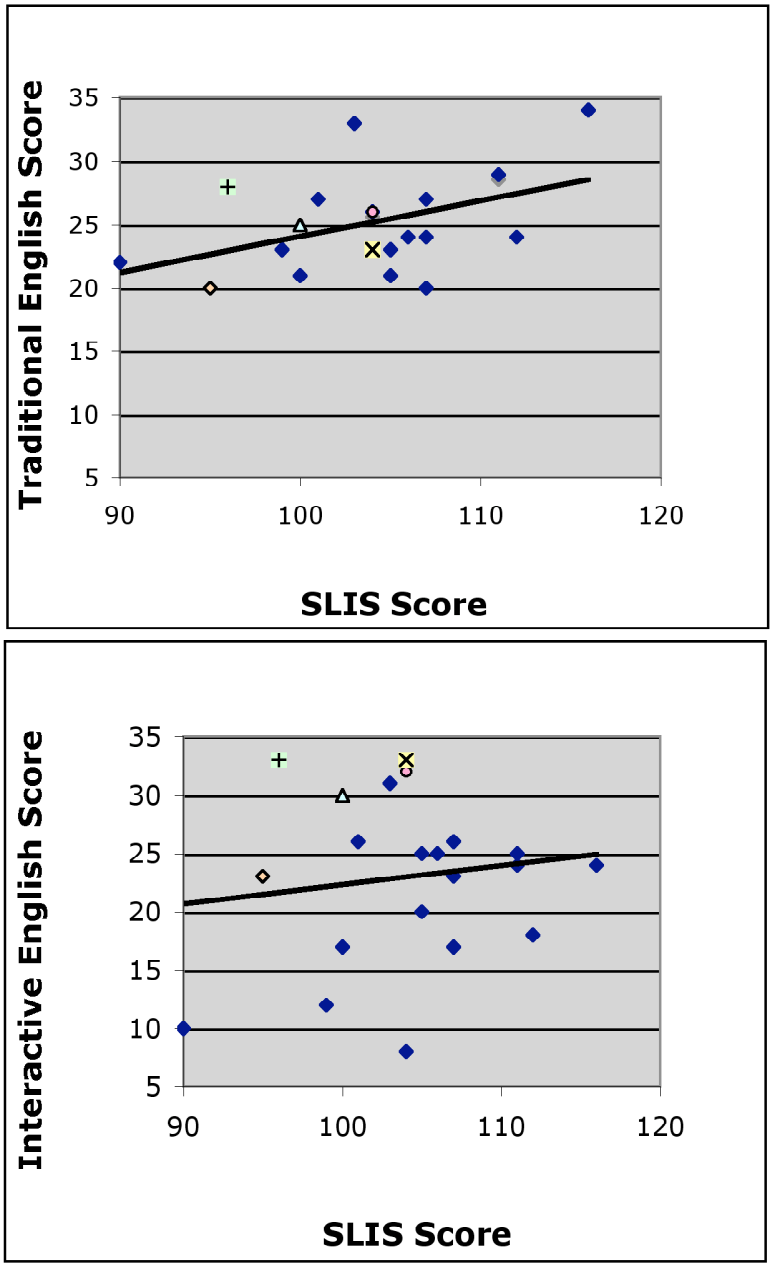

Fig. 6 Graphs of story scores versus SLIS score.

6.4 Summary of results and discussion

The study described in this section asked three important questions. The answer to the first question, "can high school students successfully write interactive stories using the ScriptEase toolset", is yes. The success rate for using both the Aurora Toolset and ScriptEase to generate an interactive story that achieves a passing IS score (at least 50%) based on the rubric of Table 1 was 73%, despite the restriction of six hours of training time.

The answer to the second question, "how do any of six factors we identified impact student performance", is that none of the six factors we measured – previous programming experience, gender, amount of previous gaming experience, amount of previous Web chatting/surfing experience, creative ability or intellectual ability – affected performance in a significant way. This result is despite the fact that there was a significant variation (large effect size) in two of these factors (previous game-play experience and previous programming experience) among study participants, particularly when males and females were considered as separate groups. Intuitively, such differences might lead one to expect that the male students would perform better at interactive story authoring than the females, since using ScriptEase for interactive story authoring has some similarity to programming and since the medium is computer games. However, this intuition is false, as determined by a direct comparison of the IS scores of the two groups.

Finally, the answer to the third question, "are there students who perform poorly relative to their peers at traditional story writing, whose relative performance improves for interactive story authoring?" is that we could not identify any such group with any degree of confidence. However, there is some weak evidence that female students with low SLIS scores could benefit the most from interactive story authoring as an alternative to traditional story writing. Whether this is the case or not would require more study. If it is the case, one potential explanation for this is that the novelty factor (females spend considerably less time playing computer games than males) may provide motivation for some students who are not otherwise motivated by traditional writing.

7. Conclusion

Researchers in the field of Computing Science are very often at the forefront in the development of new technological tools that can directly impact human problem-solving across a wide spectrum of disciplines such as Health Sciences, Engineering, and the Social Sciences (Jonassen, 2003). This study represents an important collaborative partnership between the University of Alberta's Faculties of Science and Education, two large urban Public School Boards, and a major computer games company, BioWare Corp. This research has produced a new tool (ScriptEase) that enables non-programmers (grade ten students) to author interactive stories in a computer game format. The results of this study indicate that ScriptEase offers students a viable form of creative expression through the authoring of interactive stories. ScriptEase is an environment with a significant and essential computational component that enables users to construct meaningful artifacts (interactive stories) that they can share and reflect upon (Penner, 2001).

It is important to contextualize the educational implications that ScriptEase could have for classroom teaching. Without such a tool, this level of interactive story authoring would not be accessible to students without significant computer programming training. Similarly to the way that Graphical User Interfaces enabled more users to easily and transparently interact with a computer, ScriptEase provides an analogue level of access and transparency for authoring interactive stories. A transparent computational tool that supports problem solving is a good candidate for generalizing to a larger audience (Hancock, 1995), since it allows the user to focus on the problem rather than the tool.

Many students are players ('readers') of computer games, but few are builders of computer games or authors of interactive stories. In a very real sense, the students in our study have become authors of interactive stories. Carr (2006) begins her book chapter entitled *Games and Narrative* with the following quote from her qualitative research study:

> I like RPGs [Role Playing Games] because they (normally) have a good strong story, and are normally fantasy or science fiction based, which I enjoy. Sometimes a good RPG can be like an interactive storybook. (Player interview, AP) (p. 30).

In contrast to reading stories, writing stories is a constructionist action (Papert, 1991). Learning by designing (Kafai, 1995) and subsequently building/constructing some artifact (whether virtual or physical) is fundamental to the notion of constructionist learning (Resnick, Berg, & Eisenberg, 2000). For example, El-Nasr and Smith (2006) describe two case studies, at high school and university levels, of students learning computing science skills (event-based programming, software engineering, etc.) through the development of large game design projects. They argue that students are motivated to learn computing science skills through a process they call *modding* (modifying existing games). Furthermore, they frame their educational rationale around a constructionist pedagogical approach that promotes learning by designing. In both case studies, the target student groups were programmers and the objective was to teach programming skills. These studies are in contrast to our current study where the target audience was grade ten high school English students and the objective was to have the students design and build interactive stories in the ScriptEase environment.

Bers, Ponte, Juelich, Viera, and Schenker (2002) point out that Papert's early research work on Artificial Intelligence at his MIT lab provided the impetus for four themes that form a framework for constructionist learning. The first is the belief that constructionism is a feasible educational approach that uses environments (ScriptEase) that allow students to engage in learning by design, inquiry-based lessons, and meaning-making activities. These activities result in artifacts (interactive stories) that are very often interactive, shared with others, and reflected upon by a larger community. For example, interactive stories written as part of this study can be read (played) by other students, teachers and parents, and even posted on the Web. Students can reflect on their stories in light of others' feedback and then modify their stories as they see fit. When reflection and feedback are conducted in collaboration with knowledgeable peers or instructors, scaffolding of student knowledge can support learning (Vygotsky, 1962). This has been demonstrated to occur with children even in sophisticated computational learning environments of robot building and programming (Wyeth, Venz, & Wyeth, 2004).

Papert's second theme is also illustrated here: the creation of objects enables students to work concretely and through the manipulation of these objects they learn about abstract concepts. According to Bers et al. (2002) "it is in this context that the computer, as powerful tool to design, create, and manipulate objects both in the real or virtual world" (p. 125) becomes important to education. Thus computational environments such as ScriptEase enable computers to represent and manipulate objects (text, sound, graphics, and animation), supporting Allan Kay's (Kay, 2003) observation that digital forms of media constitute a "metamedia" that subsumes all forms of media through emulation. The ScriptEase environment takes the emulation process a step further by allowing objects in a virtual story world to assume attributes that enable interactivity. Unlike written prose where the word "chest" is a symbolic representation of a physical object, ScriptEase allows the user to instantiate a virtual representation of a chest and to assign attributes that control the interaction of the reader's avatar (PC) and the chest (virtual object), based on the context of the story. The students in this study used ScriptEase encounter patterns that allow the author to specify the way in which the PC can interact with inanimate game objects, such as a chest. However, a recent addition to ScriptEase has resulted in a very important second step in allowing the artifacts in the story world to more closely emulate the imagination of the author. Behavior patterns (Cutumisu, Szafron, et al., 2006) have been added so that an author can use them to quickly specify how NPCs in the story interact with each other. The reader can now experience more vibrant bustling story scenes as the patrons of a tavern or the inhabitants of a market square interact with each other the way the author intended. Although these behavior patterns were not available to the student participants in the studies described in this paper, we intend to conduct another study to determine their utility.

The third theme of constructionism is that a student can become empowered by powerful ideas (Papert, 2000). New ideas lead to new ways of connecting knowledge and spawning further ideas. The student then begins to see linkages that weren't readily apparent before. One can assert that this is the case for any authoring process whether the student uses prose or ScriptEase to compose a story. It is the activity of creative expression that iteratively generates novel ideas, thereby empowering the individual to generate new forms of understanding. ScriptEase is simply a cognitive tool that supports and extends problem-solving techniques during the creative authoring process (Derry & Lajoie, 1993; Salomon, 1993). It does this by taking advantage of the computer gaming context that has been argued to be a powerful learning environment in and of itself (Gee, 2003). Specifically, in this gaming context it provides the author with a

collection of *proven powerful ideas* (existing patterns) and a simple mechanism (pattern adaptation) that allows these ideas to be combined and modified in new ways to support new ideas.

Finally, constructionism advocates for a learning environment that values self-reflection in which students are able to explore their thinking and understanding of knowledge. Documenting ideas and thoughts in a journal during the creative process is one method that helps promote student self-reflection about their learning. Time constraints did not allow us to introduce journal writing into the current study. Nor were we able to conduct interviews with the students. Although the quantitative results provided conclusive answers to most of the questions we asked, they also served to generate a further set of questions. These questions relate to the similarities and differences in the creative process involved in authoring interactive and traditional stories and the way that students learn to perform these activities successfully. As a result, we are currently in the process of expanding our research to incorporate qualitative research methods to address these issues. We intend to explore the ScriptEase authoring process and the traditional writing process on a personal level, examining the students' experiences/self-reflections both during and after their authoring activities. Furthermore, we are also investigating the intellectual and emotional relationships between traditional story writing and interactive story authoring.

Acknowledgement

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), the Institute for Robotics and Intelligent Systems (IRIS), and Alberta's Informatics Circle of Research Excellence (iCORE). We would like to thank the two Grade ten English classes from Edmonton high schools, their teachers, principals and computer technicians who made this study possible.

References

Anderson, C. A., & Bushman, B. J. (2001). Effects of violent video games on aggressive behavior, aggressive cognition, aggressive affect, physiological arousal, and prosocial behavior: a meta-analytic review of the scientific literature. *Psychological Science: A Journal of the American Psychological Society / APS, 12*(5), 353-359.

AAUW. (2000). *Tech-savvy: Educating girls in the new computer age*. Washington, DC, USA: American Association of University Women (AAUW) Educational Foundation Commission on Technology, Gender, and Teacher Education. Retrieved December 2, 2006, from http://www.aauw.org/member_center/publications/TechSavvy/TechSavvy.pdf.

Bers, M. U., Ponte, I., Juelich, C., Viera, A., & Schenker, J. (2002). Teachers as designers: integrating robotics in early childhood education. *Information Technology in Childhood Education (ITCE) Annual, 2002*(1), 123-145.

Bethke, E. (2003). *Game development and production*. Plano, TX: Wordware Publishing.

Bialystok, E. (2006). Effect of bilingualism and computer video game experience on the Simon task. *Canadian Journal of Experimental Psychology, 60*(1), 68-79.

Bruner, J. S. (1986). *Actual minds, possible worlds*. Cambridge, Mass.: Harvard University Press.

Bryce, J. O., & Rutter, J. (2003). Gender dynamics and the social and spatial organization of computer gaming. *Leisure Studies, 22*(1), 1-15.

Carr, D. (2006). Games and narrative. In D. Carr, D. Buckingham, A. Burn & G. Schott (Eds.), *Computer games: Text, narrative and play* (pp. 30-44). Cambridge, UK: Polity Press.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum Associates.

Cropley, A. J. (2000). Defining and measuring creativity: are creativity tests worth using? *Roeper Review: A Journal on Gifted Education*, *23*(2), 72-79.

Cutumisu, M., Szafron, D., Schaeffer, J., McNaughton, M., Roy, T., Onuczko, C., & Carbonaro, M. (2006). Generating ambient behaviors in computer role-playing games. *IEEE Journal of Intelligent Systems (IEEE IS), 21*(5), 19-27.

Cutumisu, M., Onuczko, C., Szafron, D., Schaeffer, J., McNaughton, M., Roy, T., Siegel, J., & Carbonaro, M. (2006). Evaluating pattern catalogs - The computer games experience. Paper presented at the *28th International Conference on Software Engineering (ICSE)*, China, 132-141.

Cutumisu, M., Onuczko, C., McNaughton, M., Roy, T., Schaeffer, J., Schumacher, A., Siegel, J., Szafron, D., Waugh, K., Carbonaro, M., Duff, H., & Gillis, S. (2007). ScriptEase: A generative / adaptive paradigm for game scripting. *Science of Computer Programming*, 67(1), 32-55.

Derry, S. J., & Lajoie, S. P. (1993). A middle camp for (un)intelligent instructional computing: An introduction. In S. P. Lajoie & S. J. Derry (Eds.), *Computers as cognitive tools* (pp. 1-14). Hillsdale, NJ: Lawrence Erlbaum Associates.

El-Nasr, M. S., & Smith, B. K. (2006). Learning through game modding. *Computers in Entertainment, 4*(1), 1-20.

Fischer, K. W., & Immordino-Yang, M. H. (2002). Cognitive development and education: From dynamic general structure to specific learning and teaching. In E. Lagemann (Ed.), *Traditions of scholarship in education*. Chicago: Spencer Foundation.

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

Griffiths, M.D., & Hunt, N. (1995). Computer game playing in adolescence: prevalence and demographic indicators. *Journal of Community and Applied Social Psychology*, *5*(3), 189-193.

Hancock, C. (1995). The medium and the curriculum: Reflections on transparent tools and tacit mathematics, In A. di Sessa, C. Hoyles, R. Noss (Eds.), *Computers and exploratory learning* (pp. 221-240). Berlin: Springer.

Hojat, M., & Xu, G. (2004). A visitor's guide to effect sizes - statistical significance versus practical (clinical) importance of research findings. *Advances in Health Sciences Education*, *9*(3), 241-249.

Jonassen, D. H. (2003). *Handbook of research on educational communications and technology*. Mahwah, NJ: Lawrence Erlbaum Associates.

Kafai, Y. B. (1995). *Minds in play: Computer game design as a context for children's learning*. Mahwah, NJ: Lawrence Erlbaum Associates.

Kafai, Y. B. (1996). Learning design by making games: Children's development of design strategies in the creation of a complex computational artifact. In Y. Kafai & M. Resnick (Eds.), *Constructionism in practice: Designing, thinking and learning in a digital world* (pp. 71 -96). Mahwah, NJ: Lawrence Erlbaum Associates.

Kay, A. (2003). The Dynabook revisited. A conversation with Alan Kay. *The Book & The Computer Online Symposium*. Retrieved December 2, 2006, from http://www.squeakland.org/pdf/dynabook_revisited.pdf.

Kim, K. H., Cramond, B., & Bandalos, D. L. (2006). The latent structure and measurement invariance of scores on the Torrance Tests of Creative Thinking-Figural. *Educational and Psychological Measurement*, *66*(3), 459-477.

McLellan, H. (1994). Magical stories: Blending virtual reality and artificial intelligence. In D. G. Beauchamp, R. A. Braden & R. E. Griffin (Eds.), *Imagery and Visual Literacy: Annual Conference of the International Visual and Literacy Association* (pp. 76–80).

McNaughton, M., Cutumisu, M., Szafron, D., Schaeffer, J., Redford, J., & Parker, D. (2004). ScriptEase: Generative design patterns for computer role-playing games. Paper presented at the *19th IEEE International Conference on Automated Software Engineering*. Linz, Austria, 88-99.

Minkel, W. (2002). They've got game – kids create their own computer games at the Auston Public Library. *TechKnowledge*, *48*(9), 27.

Mitchell, A., & Savill-Smith, C. (2004). *The use of computer and video games for learning: A review of the literature*. London, UK: Learning and Skills Development Agency (LSDA), Ultralab, m-learning project. Retrieved December 2, 2006, from http://www.lsda.org.uk/files/PDF/1529.pdf.

Mori, A. (2002). Video games: Bad for your brain? Retrieved December 2, 2006, from http://www.geek.com/news/geeknews/2002Jul/gam20020709015292.htm.

Murray, J. (2005). The last word on ludology v narratology in game studies. Preface to keynote talk. *DiGRA 2005 Conference: Changing Views: Worlds in Play* Retrieved December 2, 2006, from http://www.lcc.gatech.edu/~murray/digra05/lastword.pdf.

Onuczko, C., Cutumisu, M., Szafron, D., Schaeffer, J., McNaughton, M., Roy, T., Waugh, K., Carbonaro, M., & Siegel, J. (2005). A pattern catalog for computer role playing games. Paper presented at *GameOn North America*, Canada, 33-38.

Papert, S. (1991). Situating constructionism. In I. Harel & S. Papert (Eds.), *Constructionism* (pp.1-11). Norwood, NJ: Ablex.

Papert, S. (2000). What's the big idea? Toward a pedagogy of idea power. *IBM Systems Journal, 39*(3), 720-729.

Penner, D. E. (2001). Cognition, computers, and synthetic science: Building knowledge and meaning through modeling. In W. G. Secada (Ed.), *Review of research in education* (pp. 1–35). Washington, DC: American Educational Research Association.

Resnick, M., Berg, R., & Eisenberg, M. (2000). Beyond black boxes: bringing transparency and aesthetics back to scientific investigation. *Journal of the Learning Sciences*, *9*(1), 7-30.

Rhyne, T. M. (2002). Computer games and scientific visualization. *Communications of the ACM, 45*(7), 40-44.

Robertson, J., & Good, J. (2005). Story creation in virtual game worlds. *Communications of the ACM*, *48*(1), 61-65.

Salomon, G. (1993). On the nature of pedagogic computer tools. The case of the writing partner. In S. P. Lajoie & S. J. Derry (Eds.), *Computers as cognitive tools* (pp. 179-196). Hillsdale, NJ: Lawrence Erlbaum Associates.

Szafron, D., Carbonaro, M., Cutumisu, M., Gillis, S., McNaughton, M., Onuczko, C., Roy, T., & Schaeffer, J. (2005). Writing interactive stories in the classroom. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning (IMEJ), 7*(1).

Tompkins, G. E. (1982). Seven reasons why children should write stories. *Language Arts*, 59(7), 718-721.

Torrance, E. P. (1999). *Torrance test of creative thinking: Norms and technical manual*. Beaconville, IL: Scholastic Testing Services.

Vygotsky, L. S. (1962). *Thought and language*. Cambridge, MA: MIT Press.

Wyeth, P., Venz, M., & Wyeth, G. (2004). Scaffolding children's robot building and programming activities. In D. Polani, B. Browning, A. Bonarini, & K. Yoshida (Eds.) *RoboCup 2003: Robot Soccer World Cup VII, Lecture Notes in Computer Science,* Vol. 3020 (pp. 308-319). New York, NY: Springer-Verlag.

Yelland, N. & Lloyd, M. (2001). Virtual kids of the 21st century: understanding the children in schools today. *Information Technology in Childhood Education Annual, 13*, 175–192.

Young, B. J. (2000). Gender differences in student attitudes toward computers. *Journal of Research on Computing in Education*, 33(2), 204 - 216.

Zachary, R. A. (2000). *Shipley institute of living scale, Revised Manual*. Los Angeles: Western Psychology Services.

Zachary, R. A., Crumpton, E., & Spiegel, D. E. (1985). Estimating WAIS-R IQ from the Shipley institute of living scale. *Journal of Clinical Psychology*, *41*(4), 532-540.