Data Driven Storytelling for Digital Learning Games

Extended Abstract

Ursula Wolz
Bennington College
1 College Drive, Bennington VT 05257
USA
ursula.wolz@gmail.com

Zachary Segall
Grinnell College, Computer Science
Grinnell, IA, 50112
segallza@grinnell.edug

Gemma Nash
Grinnell College, Computer Science
Grinnell, IA, 50112
nashgemm@grinnell.edug

ABSTRACT

This[1] poster presents a new theoretical framework for developing worlds for virtual novels and other such story-based games. Over the past half a century, games have held tremendous promise for widespread educational opportunity, but have yet to see massive scale success. In the process of developing *Genderless in Germany,* an educational virtual novel with an embedded individual tutoring system, we encountered the challenge of encoding content for the world behind the game and how to connect it with the tutoring model.

CCS CONCEPTS

Applied computing ~ Interactive learning environments; *Computer-assisted instruction*

KEYWORDS

Tutoring; educational games; data-driven design

1 INTRODUCTION

Computer-based games have held promise for education for almost 50 years. While game technology has progressed significantly with immersive worlds, rich graphics, and sophisticated interactivity, there is a continuing struggle to provide opportunities for highly personal and authentic learning that can be efficiently assessed, all within a compelling experience. Games for learning provide rich microworlds[Papert 1979] and opportunities for augmented learning [Klopfer 2008]that promote discovery-based problem solving. But students can get stuck puzzling out a solution, when a simple hint in the form of a teachable moment, could send them on their way. Moreover, current practice in educational games [Prensky 2005, Taylor 1980] is to correlate learning outcomes with out of game traditional(test-based) protocols, rather than to create fully self-contained learning experiences that draw on the strengths of the computer games. In this poster we will present an architecture for data-driven representation storytelling and background world model to support pedagogical goals in a game.

Mobile devices are opening new possibilities for all-the-time, on-demand education. The vision of a tutor in your pocket is becoming a reality through existing Internet-based technologies, where compelling interactivity is networked with remote sophisticated analysis systems. Currently, these systems remain primarily vehicles for reinforcement for delivered instruction; the student completes an assigned problem in order to practice a skill that was taught through formal instruction[Taylor 1980]. Similarly, online tutoring systems, and Massively Online Open Classroom technology (MOOCS) are increasingly using machine learning and data science to infer learning outcomes based on the final product produced by the student rather than considering how the student got there[MOOCWksp].

The *Genderless in Germany* educational game project explores how intelligent tutoring technology can be imbedded in video games. Our hypothesis is that one way the separate limitations of educational games and intelligent tutoring systems can be overcome is by combining technologies: a tutoring system can provide the individualized feedback necessary for a learner to progress through difficult points in a game, while a well-built education game creates an engaging environment in which the student can learn. Our goal is to define architectures that provide data-driven game development techniques for implementing the game play[Gregory, 2009]. In game development, data driven refers to game components such as graphics and sound that are represented as assets, which is the data fed to a suite of engines. Game development environments such as Unity3D employ engines for game components such as sound, 3D rendering, animation, interaction and physics, so that a significant amount of development occurs without a programmer.

---

*Produces the permission block, and copyright information
†The full version of the author's guide is available as acmart.pdf document
[1]It is a datatype.

WOODSTOCK'97, July 2016, El Paso, Texas USA

DOI: 10.1145/123_4

After the development of an initial prototype for the *Genderless in Germany* educational game project, we created a theoretical framework to help organize the coding framework for a data-driven game development system. Our framework splits the content the developers need to create into three sections: a world model, narrative model, and tutoring model, each of which control different dimensions of the game experience and are modular components which can be developed separately, by designers with little coding expertise.

2 GENDERLESS IN GERMANY

GenderlessInGermany (GiG) is an enhanced text-based adventure game – a visual novel with animations that teaches German vocabulary and grammar using Total Physical Response(TPR) theory[Ault et al. 2009]. The player is an alien from an androgynous culture who must master the nuance of German articles, vocabulary and grammar to survive. All interaction to move forward in the game occurs through the textbox. The graphics and animations only provide contextual information about the world where the animation is used to support TPR. The genre and domain were chosen because (1) text-based games are relatively easy to implement but also requires a more robust backstory, challenging our question of how to separate the world model and tutor,(2) natural language learning provides a structured procedural task (e.g constructing a successful utterance that is less structured than mathematical transformations, but still sufficiently structured to allow rapid development of the world model, (3) there isn't one right answer in a communicative act, providing a test bed for generating solutions rather than precoding the solutions [Wolz 1989]. German was chosen because of its systematic and consistent grammar, and because it is a second language of two of the current game developers. We were also looking for a game that would provide comedic relief rather than violence as a motivator. The absurdity, and therefore comedic potential, of is gendered nouns was addressed by none other than Mark Twain.

2.1 Initial Prototype Implementation

For the initial prototype for GiG, we focused primarily on building up a simple framework with which to test the data driven world model and story-telling aspects of the game design. We built the prototype around the first level of the game - where the alien crashes the space ship, leaves to explore the surrounding area, and encounters a helper who starts teaching them German. In the reduced prototype, all commands are given in English. We focused on the issue of how to manage the effects of the player's interactions with the world and how to present the narrative events. The entire prototype was implemented in Unity and is designed so that entire levels can be created working only through the Unity editor, so that no knowledge of coding is necessary to develop content. The initial prototype did not include the tutoring model which we anticipate will be an overlay to the storytelling architecture.

2.2 The World Model/Narrative Model Framework

In order to make building a story more tractable, we designed a conceptual model that splits the necessary information for managing a story driven game into two parts: the world model and the narrative model. The world model keeps track of the state of the world, which can be organized by scenes or locations. The world model is object oriented, where everything relevant element of the world the player is operating in is represented in some object, and interactions are governed by a rule base. By building general templates for objects, the game designer can reduce overhead for building specific levels - the designer can draw objects from a general pool and effectively place them in a scene, rather than defining the properties for each object in a scene from scratch. The world model governs and manages the impacts of a player's actions on the game world, even if the actions aren't immediately visible to the player.

The narrative model is layered on top of the world model in order to incorporate the storytelling aspect of the game. The narrative model is where the story that drives the game is captured. Where the world model codifies relevant items and their interactions into objects, the narrative model codifies a story into discrete narrative events. Narrative events can be arranged into a tree which progression is governed by the player taking certain actions within the world model, or into a more nonlinear form, where a narrative event is involved based on the state of the world model. Narrative events can be triggered when changes within the world model reach some critical mass, and it makes sense for the story to progress based on such changes. The narrative model provides the primary source of information for the player, giving them the story that motivates the game.

2.3 Data Driven Game Development

The conceptual separation of the world model and the narrative model creates a new perspective on the type of assets necessary to support data driven game development. While both the behavior of the world and the story are the same type of data from a computer perspective, i.e. both are types of text and are often programmed directly into the game, they can be quite distinct from the developer's perspective: the rules and objects of a world can operate near independently from the story taking place in the world. Separating the types of content out from each other makes it easier to build engines to parse objects and rules versus story events. Additionally, the rule based nature of the world model may make it possible to automate the generation of the world model, given a relatively small stating set of rules. In contrast, it may never be possible to automate generation of story, since each interaction needs to be appropriately tailored to the character's situation.

2.4 Implications for Gameplay

The world model/narrative model split also allows the developer to give the player significantly more freedom in action without creating additional overhead for the developer: in many older adventure games, the developers had to code each possible action the player could take by hand, as well as the effects of the actions in each possible circumstance. Adding more choices would lead to an exponential increase of content the developer would have to implement. The world model addresses this issue by having player's actions be translated into effects that are governed by the rules and objects in the game world - implementing a handful of rules is much less content intensive than programming all possible states the player could be in. In the world model alone, the player has a wide variety of actions they can take and corresponding effects; the narrative model gives the developer the ability to control the player's progression through the story. By having a relatively small number of narrative events which are triggered by certain states of the world, the developer does not have to worry about the series of choices that leads a player to a state, only what narrative events should correspond with a certain state of the world and where they want the player to be able to progress from that world state.

2.5 Extension into the Tutoring Model

The world model/narrative model can also be extended along a third dimension for educational games, into a tutoring model. The tutoring model is based around a progression of concepts that the student needs to learn. In order to provide a challenge, the correct progression of concepts also needs to be accompanied by incorrect interpretations of important concepts, called distractors. The tutoring model can be incorporated into the other two models. Correct manipulation of the information in the given field of study can be made a requirement for the production of positive effects in the world model. Positive or negative feedback can be given through the narrative model, to the point where winning states correspond to demonstrated mastery of concepts. At all stages, narrative events can be interlaced with information that is relevant to the subject at hand.

3 DISCUSSION

This work in progress will benefit from review via a poster presentation. We continue to implement data structures and game objects in Unity3D to support both the world model and storytelling architectures. An extention that we are exploring this summer to provide a statistics based world model implemented via dynamic programming.

ACKNOWLEDGMENTS

This work was supported by the Noyce Visiting Professorship at Grinnell College in Academic Year 2016-2017.

REFERENCES

[Ault et al. 2009] C. Ault, A. Warner-Ault, U. Wolz, T.M. Nakra 2009 “Kinesthetic Communication for Learning in Immersive Worlds.” in *Serious Game Design and Development: Technologies for Training and Learning,* J. Cannon-Bowers and Bowers C., (Eds.) IGI Global, Hershey, PA

[Gregory 2009] J. Gregory, 2009 *Game Engine Architecture,* Taylor & Frances

[Klopfer 2008] E. Klopfer 2008 *Augmented Learning,* Cambridge: MIT Press.

[MOOCWksop] http://www.moocworkshop.org/

[Papert 1987] S. Papert. 1987, Mindstorms: Computers, Children and Powerful Ideas, Basic Books

[Prensky 2005] M. PRENSKY. 2005 Computer games and learning: Digital game-based learning. *Handbook of computer games studies,* 18:97-12.

[Taylor 1980] R. TAYLOR. 1980 *The Computer in the School: Tutor, Tool, Tutee,* Teachers College Press

[Wolz et. al. 1989] Wolz, U., McKeown, K. R., and Kaiser, G. E. 1989. Automated tutoring in interactive environments: a task-centered approach. Mach. Mediat. Learn. 3, 1 (Jan. 1989), 53-79.