Authoring tool architecture for edutainment systems

Gilberto Huesca[1] and Julieta Noguez[1],

[1] Tecnológico de Monterrey, Campus Cd. de México, Calle del Puente 222, Col. Ejidos de Huipulco, Tlalpan 14380 México, D. F., México
{ghjuarez, jnoguez}@itesm.mx

(Paper received on September 07, 2010, accepted on September 28, 2010)

Abstract. Pedagogical theories have embraced games in the past years as a teaching tool. This is because games have an intrinsic entertainment characteristic that can be used as a vehicle to deliver educational concepts and to engage students in the teaching-learning process. Edutainment, as a special type of serious games, is nowadays used as a way to engage students and improve education delivery. But, building an edutainment system is a complex task because it is needed a hybrid work team. Design, art, developers and pedagogical experts are required to work together to create the serious game. Also, professors, that will apply this tool into the class, must have an active role in the design process to adapt the game to their specific needs. Authoring tools can help in this process by providing professors with instruments that require less computer expertise in order to create a serious game of this kind.

Keywords: Authoring tools, Edutainment, Serious Games, System Architectures

1 Introduction

Engaging capabilities of games can be used as a medium to expose educational concepts so students can learn with less resistance. Computational research over years has made available new developments so new platforms can be used to build more entertaining and funnier games. Also, large companies have worked to distribute these systems to consumers directly to their home.

From another point of view, new generations are more used to use this kind of technologies. Nowadays, young people are not afraid of using cell phones as an entertainment tool rather than only a communication device for example. Also, these systems are more and more popular and have a great acceptation between people.

Serious games have emerged as a computational application that takes profit from these elements and combines serious aspects (teaching, learning, communication or information) with the playful characteristics of video games [1]. As an educative tool, edutainment (contraction of education and entertainment) is a specific serious game that combines education and entertainment. Edutainment is a methodology that combines teaching methods and games characteristics to engage students in order to make easier their learning process [2]. It tries to increase the educational value of games by adding pedagogical techniques to display educational content. A main objective is to present education in a less stressful way as it can be in traditional ways.

©G. Arroyo-Figueroa (Ed)
Special Issue in Advances in Artificial Intelligence and Applications
Research in Computing Science 51, 2010, pp. 139-146

But creating such an educational tool is not an easy task. Edutainment systems development needs a hybrid development team. A design team is required to work over the story line. Also, an art team must get involved to create all game images and sounds. A software development team will put all this together to have a final product. But, as an educational tool, a pedagogical expert is necessary too. The professor that will use it must have an active role in this process. But not always these professors have the computational expertise to compose the system.

Authoring systems are a framework that allows virtual learning environments' production with a lower material (money and time) cost and that reduce the abilities or training needed to create them [3]. These tools can be used in order to help professors to get involved in the development process for edutainment systems.

In this work, we propose an authoring tool architecture that will help such a hybrid team to design and implement an edutainment system and that will aid the professor to get involved into it.

2 Related work

As stated before, the construction of an authoring tool requires integrating different components that require different knowledge to be developed. There are some works that try to solve the task of developing some components like learning environments or to compose multimedia applications. These tools, put together, can be useful to create an edutainment system. Next, we will describe some of these tools.

2.1 Adaptive Lecture Authoring Module

This architecture describes four components [4]: Teacher profiler, Student profiler, Lecture profiler and Notification manager. This four components work together in order to create profiles that will be cross referenced to give the student a notification of a new lecture that is available in the system and that fulfill the students' learning needs. The Teacher profiler allows the teacher to define properties such as name and assigned courses. Through the Lecture profiler, the teachers can create learning content in the formats they need for their courses. They may also assign metadata to the lectures (such as the level of difficulty) to help the adaptation to the students needs. The Student profiler allows the definition of student general information (name or enrolled courses) and the results of quizzes taken in the system. The Notification manager will analyze this students' information and the characteristics of the lessons to find the best material that specifically help students in their learning process.

This architecture highlights the need for two aspects within authoring tools' architecture. One of them is the existence of components that help the teacher to define the data that will characterize the courses, lectures and students. This information will help the system to give students a specific learning content to help their learning process. On the other hand, components that help the professors to design and create these learning contents are needed. In an edutainment system, this component must help the professor to create the digital components that will compose the serious game like animations or sounds.

2.2 Architecture for authoring tools based on interactive television services

Kuo-Shu Hsu et al [5] argue that traditional authoring tools for sequencing of events are based on a linear representation of the content sequence. They propose a hierarchical model composition of scenes. The scenes sequence is represented by a tree. Each node represents a scene and each path, a sequence of scenes that responds to the player's interactions with the components of each scene. The root of the tree is the initial stage of the system. Each scene is a container for artifacts. An artifact is a digital item which can be static or that can allow interaction. Under this idea, the authors present the architecture for authoring tools based on interactive television services. The main component of this architecture is the Author controller. Its job is to coordinate the other components interactions. Through a graphical user interface (GUI) editor component, professors can create the structure of the scene to define the layout and the artifacts in it. This information is stored in a repository in an XML format. When a component is requested to deploy, the XML[1] is retrieved from the repository and this is translated into code to deploy the artifacts to the student.

This architecture focuses to two aspects. The first one is the presence of components that help to manage the creation and persistence of scenes and artifacts. The other one is the existence of components that help the use and deployment of these artifacts. This is very important for edutainment environments because professors must be provided with simple GUI tools to manipulate digital content to compose the game. Also, the definition of formats to store these digital contents, so to be displayed after, is required in order to facilitate its management.

2.3 Authoring tool for the composition of 3D audiovisual scenes

The main contribution of this architecture [6] is the management of objects created by the authoring tool. The architecture components are dedicated to the persistence and recovery of material. This architecture is intended to help video makers to open, save or play them inside a learning context. Despite being a simple architecture, it highlights again the need of internal formats that will help to manage and display the learning contents and other·components that will manage the author interactions.

3 Edutainment architecture

As a semi open learning system[2], edutainment requires a high interaction from students. These interactions must be assessed to infer the students' cognitive state and to give them specific tasks to do in the game to give support to their specific learning

[1] XML stands for Extensible Markup Language. It is a text format originally designed to meet the challenges of large-scale electronic publishing, XML is also used as a language for exchange data.

[2] An open learning system is an environment that helps students to interact and explore learning environment resources by a subset of parameters in order to obtain a better comprehension of the experiments and concepts that are exposed [7]. In a semi open learning environment an educative goal is added in order to infer students knowledge[8]

needs. [9] proposes a generic architecture for this kind of environments. This architecture proposes three components: A User Interface component, an Intelligent Tutoring System and a Virtual environment component (Fig. 1). By means of the User interface, students interact with the Virtual environment. These interactions are collected and sent to the Intelligent Tutoring System to be analyzed and to give back, to the students, lessons or quizzes to help them in their learning process.

Based on this architecture, we propose an edutainment systems architecture. Fig. 2 presents the RchEd architecture.

The User Interface component will display the multimedia content and will capture students' interactions. The Edutainment System component will manage the game mechanics and will contain a simulation module to construct multimedia models. This component will also gather information from students' interactions. These interactions will be useful to apply an inference process to track the students' cognitive state. The Intelligent Tutoring System will apply this inference process to the data collected by the Edutainment System. The results obtained by this procedure will be fed back to the Edutainment System component in order to modify the story line. This modification will present the students with new challenges where they will be able to continue the learning process or rectify their answers. It is important to see that these results are presented by means of the simulation and not directly to the student as a test or lesson to learn like in the semi open learning environment architecture of [8].

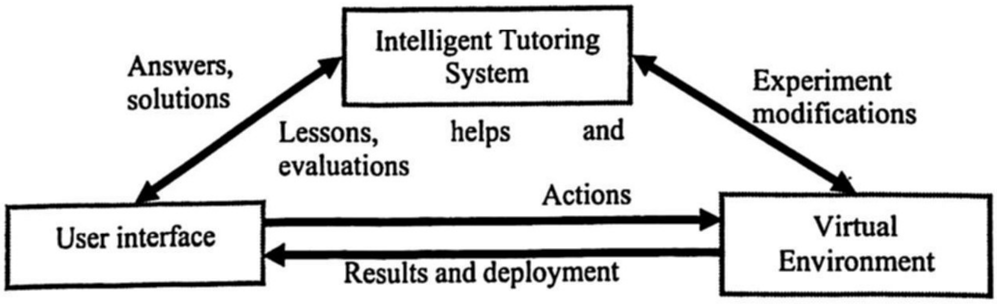

Fig. 1. High level view for a semi open learning environment architecture [8]

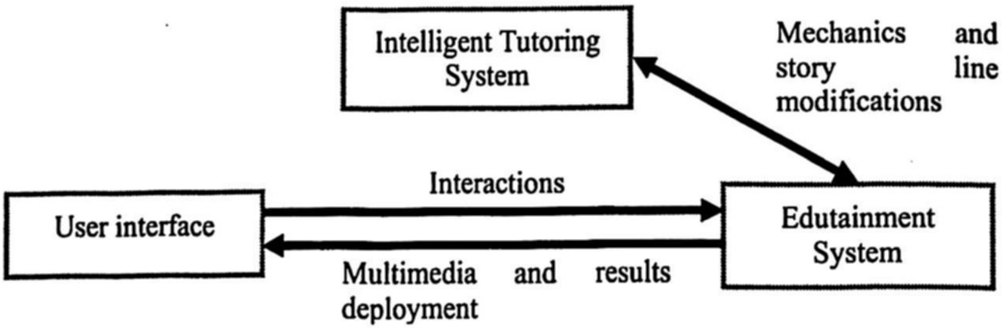
Intelligent Tutoring System
Mechanics and story line modifications
Interactions
User interface
Edutainment System
Multimedia and results deployment

***Figure.* 2.** *High level view for the RchEd architecture for edutainment semi open learning systems*

This is because in edutainment systems, education is presented within an entertainment goal. By not communicating directly the Intelligent Tutoring System decisions to the students, the entertainment goal is not affected and the students enjoy more the interaction with the system. This is helpful because the more the student interacts with the system the more the student can acquire the knowledge and abilities exposed in the learning environment. Also, by modifying the storyline components according to the Intelligent Tutoring System, the entertainment goal can be enriched with hard challenges or more appealing plots.

4 Authoring tool architecture

Edutainment systems focus their attention into two elements: education and entertainment. This can be seen in the RchEd architecture that defines a component to manage education (Intelligent Tutoring System along with the collaboration of the other components) and another component to manage entertainment (Edutainment System along with the collaboration of the other components). Like it has been seen in the authoring tools described, it is very important that authoring system architectures define components to help professors by defining elements for these to aspects. The Archaud authoring tool architecture proposed in this work (Fig. 3) has two components that will help professors to center into these two features and combine them to secure the educational and entertainment aspects in the final game.

The Knowledge Arguments component focuses its capabilities to provide authors with tools to define the domain structure to be taught. Within this component, professors can define which educational components will be displayed, when and where to be displayed. Also, the professor is able to define the inference structure that will be used by the intelligent tutoring system to evaluate the students' interactions to know the actual students' cognitive state. This component also allows the definition of the pedagogical actions that will be performed after this process is done. These pedagogical actions can be the repetition of the same or a similar challenge with less or more difficulty as reinforcement, or to present a challenge where the student needs to use the last knowledge obtained to solve a puzzle. These pedagogical actions must be aligned with the main storyline in order to show progress in the game.

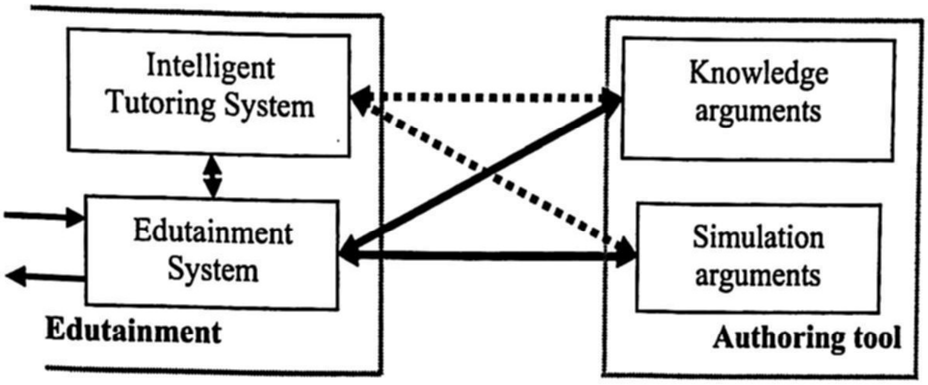

***Figure.* 3.** *High level view of the Archaud architecture for authoring tools for edutainment systems*

All these definitions need information from the game multimedia objects configurations because these actions will affect the multimedia deployment of the game. Due to this situation, this component must communicate with the Intelligent Tutoring System component from the Edutainment Environment and also with the Simulation Arguments component from the Authoring Tool.

On the other hand, the Simulation Arguments component helps with the definition of the entertainment characteristics of the serious game. This component helps to define the multimedia objects that compose the game. Also, the professor can define which multimedia objects will be displayed, when and where. Professors can also define which parameters, from multimedia objects, will be used as information to be fed to the inference model. That is what students' actions will be stored, how this information is collected, when the inference process will be started and how results will be displayed to the students.

Fig. 4 shows a specification view of the Archaud architecture and the relationships with the RchEd architecture.

The Knowledge items relationships module will hel top define the model used to represent the domain model for the Intelligent Tutoring System. By means of it, the professors can define the knowledge items that they want to teach and their relationships. This model should be closely related to the topics in which the edutainment system will be applied. This representation may be a concept map showing where the components of knowledge and its hierarchy. This module should provide the necessary tools for the author to describe each of these elements and to relate them.

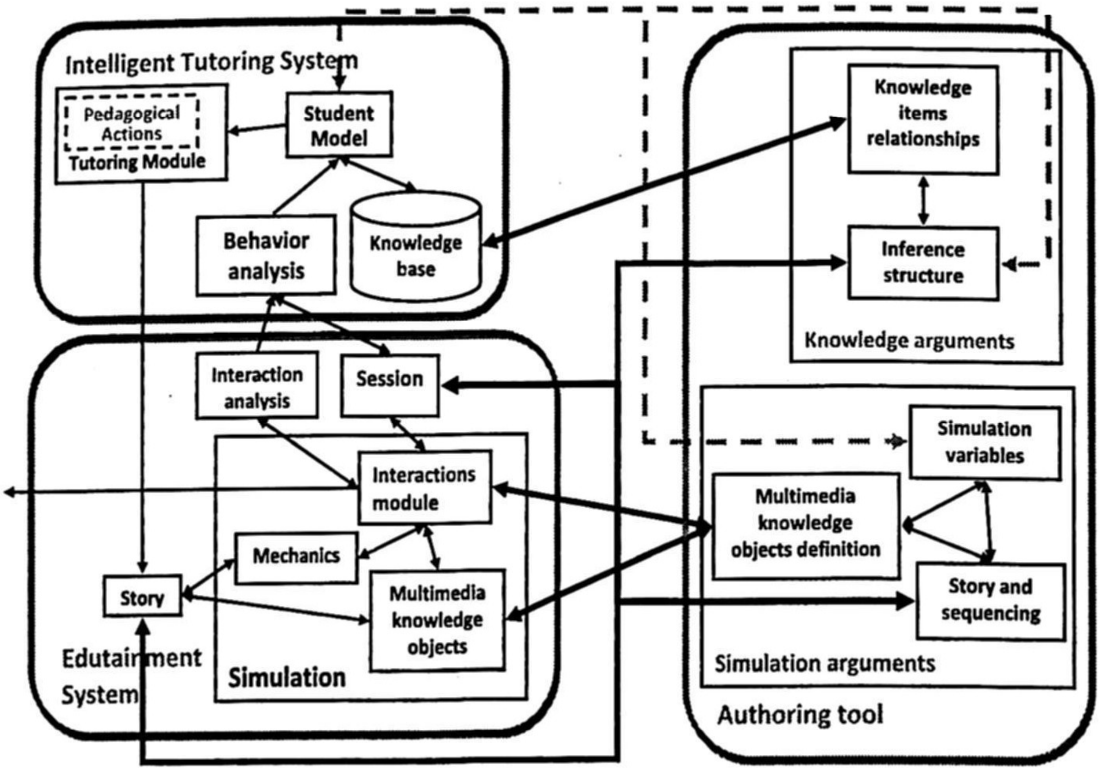

Figure 4. *Archaud architecture (right side) for authoring tools for edutainment systems and interactions with RchEd architecture modules (left side)*

The Inference structure module works over the structures de defined in the previous module to define a student model structure for the Intelligent Tutoring System. This means that the professor should be able to indicate which components of the global domain model affect each of the scenes or events in the game that require a process of inference. Based on these models, the module will enable the professors to define the way in which they perform the inference of the new cognitive state. For example, the author may indicate the weights of the relations between knowledge items to focus the inference to a specific aspect. To have all the information required for this inference process, the module will allow the identification of elements that affect the student model, such as students' interactions.

The Multimedia knowledge objects definition module allows the authoring for the digital components to be deployed in the serious game. Professors can allow individual modification and customization of the elements to be used. This customization would include the modification of state and behavior of the objects. This module can also allow the composition of objects.

The Story and sequencing module helps to define the main plot of the game. By means of this module, professors should be able to define the sequence of events in the game and how these events start or end. Also, the module can help the professors to define the relationships between digital objects used in the learning environment and their sequence of deployment in the simulation (i.e. if one object can start or stop another object action).

The Simulation variables module is primarily responsible for defining the simulation variables that will affect the inference structure. This means that the professor can define variables in the simulation (object state, student interactions, timing, etc.) that may be used in the structure used to infer the new student cognitive states. Some information needed in the inference process is obtained from the simulation models and objects, and vice versa, the results of this inference process will affect directly the digital deployment and the storyline of the game. Due to this, these two components of the Archaud architecture must work together.

5 Conclusions and future Work

Based on the RchEd architecture, we are working in a framework that will help in the implementation and configuration of edutainment environments (Fig. 5). The framework shows the composition of a scene by events and a game by scenes. Scenes and Events implement the interface Connectable in order to allow professors to connect the same event or scene to different ones in different games. An event could also implement the interface Scorable to score students interactions. An Intelligent Tutoring System could be useful in this process. This framework will be also the foundation for a further implementation of an authoring tool for edutainment systems based on the Archaud architecture. This will help to implement some events that will be used to compose a Physics RPG (Role Playing Game) game. The authoring tool will allow professors to define the game events, the configuration for these events and the event and scene sequences in the game. Once finished this work, the game will be tested in a pretest-posttest process that employs leaning gains in order to know if such a game and architectures help in the learning-teaching process. This process will be applied to an undergraduate group in a Physics course.

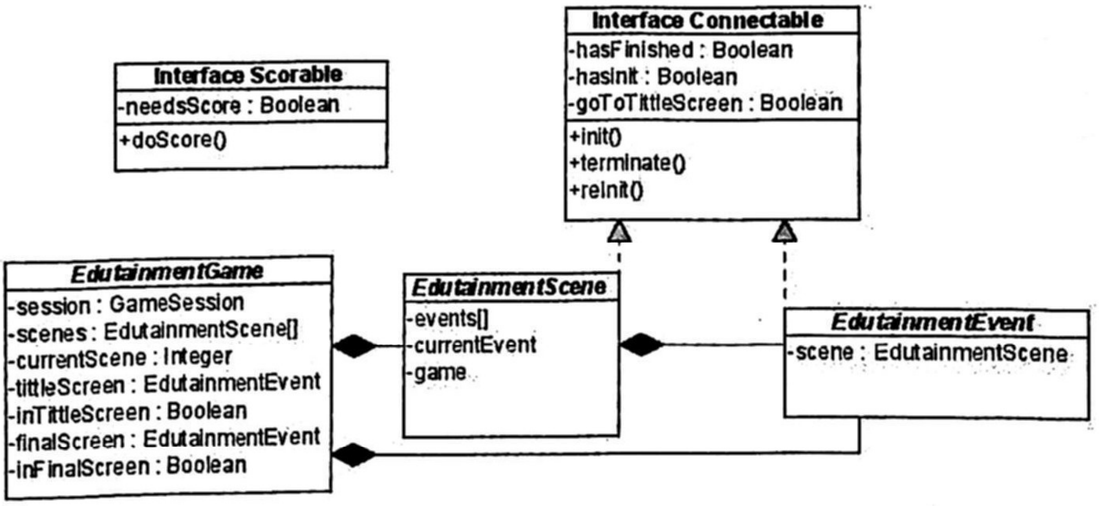

Figure. 5. Framework to build edutainment environments designed under RchEd and Archaud architectures.

References

1. Álvarez, Julián. DU JEU VIDÉO AU SERIOUS GAME: Approches culturelle, pragmatique et formelle. Tesis Spécialité science de la communication et de l'information. LARA Université Toulouse le Mirail. 2007. PhD thesis
2. Qianping Wang; Wei Tan; Bo Song; Research and Design of Edutainment Information Technologies and Applications in Education, 2007. ISITAE '07. First IEEE International Symposium on. November 2007. pp 502-505
3. Murray,T. MetaLinks: Authoring and affordances for conceptual and narrative flow in adaptive hyperbooks. Internacional Journal on Artificial Intelligence in Education. Vol. 13. pp. 199 - 233.
4. Khalid, S.U.; Basharat, A.; Shahid, A.A. y Hassan, S. An adaptive E-learning Framework to supporting new ways of teaching and learning. Information and Communication Technologies, 2009. ICICT '09. International Conference on. August 2009.
5. Kuo-Shu Hsu, Chyan Yang y Chao, T.-H.S. An Authoring Tool and Extendable Production Architecture for Interactive Television Services. Consumer Electronics, 2006. ISCE '06. 2006 IEEE Tenth International Symposium. 2006.
6. Daras, P., Kompatsiaris, I., Raptis, T., y Strintzis, M. G. MPEG-4 Authoring Tool for the Composition of 3D Audiovisual Scenes. On Proceedings of the Second international Workshop on Digital and Computational Video. February 2001
7. Shute V. Glaser R. A Large Scale Evaluation of an Intelligent Discovery World. Smithtown. Interactive Learning Environments 1. pp. 55-77. 1990
8. Noguez J., Sucar E. (2005) A semi-open learning environment for virtual laboratories. Lectures Notes in Artificial Intelligence 3789. Springer-Verlag. 2005. ISSN 0302-9743. Pp.1185-1194
9. Noguez, Julieta. Modelo probabilista relacional del estudiante para laboratorios virtuales. Tesis doctoral. Instituto Tecnológico y de Estudios Superiores de Monterrey campus Cuernavaca. 2005. PhD. Thesis.