Simulation, Games, and Virtual Environments in IT Education

Norman Pendegraft
University of Idaho, USA

INTRODUCTION

The rapid change in information technology presents several problems to IS educators and trainers. In particular, the number of concepts that must be mastered is constantly increasing while the time available is not. This makes it essential to use class time efficiently as well as effectively. Simulations and games provide interesting and useful tools to help in this effort.

BACKGROUND

The idea that students learn better by doing goes back at least to Dewey (1938). The key idea underscoring this approach is that people learn better from experience than from reading or listening (Corbeil, Laveault, and Saint-Germain, 1989). This sort of experience can be gained in a simulation or game. By compressing time, the simulation allows the students to experience the consequences of their own actions or to see how a system operates.

Simulation, case studies, role playing, and gaming are related teaching methods based on experiential learning. They permit experience or experimentation with a situation modeling the real world (Senge, 1990). On a deeper level, simulation is claimed by some to be a fundamentally new way of studying the world (Pagels, 1998). Narayanasamy, Wong, Fung, and Rai (2006) distinguish between games, simulation games, and training simulators. Simulators are models used for systems analysis or policy formation. Simulators may use mathematical models and Monte Carlo or discrete event methodologies. They argue that training simulators offer real-world environments and challenges are focused on skill development rather than entertainment, and are not goal oriented. Klassen and Willoughby (2003) discuss the importance of assessment and present data supporting the notion that games help students learn more quickly than do lectures.

Case studies are a time honored approach of instruction in strategy courses (see, for example, Burgelman, Maidique, and Wheelwright, 2001). Barker (2002) suggests that they can also be very valuable for teaching technical skills such as software development. In some sense, a case study is a role play with the student acting the part of an analyst examining the case situation.

Role playing and simulation gaming are similar approaches in that they use simulated worlds, but instead of creating, or observing or analyzing that world, students are immersed in it. Role playing is a method in which students are presented a scenario simulating some real situation, and assigned roles in that scenario. The scenario can be based on real or simulated situations (Barker, 2003). Participants then assume the roles of relevant persons in the scenario and act out the situation to see what happens. Role playing is a commonly and successfully used tool in IS education (for example, Christozov, 2003).

According to Greenblat (1988), simulation gaming includes role playing as an element. Whereas role playing allows participants to play the roles as they please, simulation gaming emphasizes the interactions of the roles and constraints of various types on the players. In some sense, a simulation game strives to teach about a specific situation while a role play or game may have a more general lesson.

Greitzer, Kuchar and Huston. (2007) describe cognitive principles for learning and a process for designing and improving game based education. In particular, they argue that experience should be presented in realistic contexts. They identify features of games that attract extended play including levels, adaptability, clear goals, interactions, and shared experience with others. They describe the use of these principles in a security training game.

The use of modern information technology in developing these games and simulations, stimulated, no doubt, by the vibrant computer game industry, has led several authors to create "virtual" environments for training. This seems a good umbrella term to include all simulation and gaming approaches to training, as they do indeed create virtual realities in which the students operate. Summerfield (2004) suggests that role playing is superior for learning soft skills (like dealing with people) while technically based simulations are useful for learning hard skills.

Simulation and gaming have been used to enhance training in a variety of non IS areas including incident management (Jain and McLean, 2003), mass casualty medicine (Müller, Martens, Willen, and Müller, 2000), military technology

Copyright © 2009, IGI Global, distributing in print or electronic forms without written permission of IGI Global is prohibited.

(Meeds, 2001), military tactics (Chatham, 2007), and immunology (Kelly, Howell, Glinert, Holding, Swain, Burrowbridge, and Roper, 2007). Mayo (2007) argues that video games can be used to teach science and engineering better than lectures.

After many years of using such exercises at all levels (undergraduate, graduate, and executive), it is the author's opinion that they are very useful and that major benefit accrues to the instructor in preparing the simulation as well as to the students when they play the game. Simulation and gaming are student centered learning, that is, the student is actively involved in the learning rather than passively observing the instructor (Greenblat, 1998). The student does the work, makes decisions and sees the impact of the decisions. Role playing and simulation gaming attempt to take advantage of this by creating a situation in which a student may "play a game" in which time is compressed and attention can be focused on a few key ideas. Finally, these kinds of exercises are fun. The class gets to move around, talk, and frequently laugh. Simulations and games epitomize the idea that learning should be fun.

SIMULATION AND GAMING IN IT

Simulation as a teaching tool suggests several approaches. Perhaps most obvious in an information systems curriculum is computer simulation. Using this technique, a computer program is written which exhibits behavior that models the behavior of the system under study. Butterfield and Pendegraft (1998) described a spread sheet simulation of a Fourier Series, adding sine wave to construct a square wave for a class demonstration of the impact of bandwidth limits on data rates. Campbell (1996) created a simulation of a computer and had his students write assembly language programs to execute on the simulation. Zant (2001) used a computer simulation of a CPU for in class demonstrations and for homework problems. Englander (2003) used Little Man Computer, a simple paper simulation of a CPU, as an example to explain basic CPU architecture, CPU operation, and machine language. In an extension of those ideas, Pendegraft and Stone (2003) had their students develop a Visual Basic simulation of a Little Man central processing unit on which they ran programs mandated by the instructor. In addition to having to execute simple programs written in Little Man's machine language, their simulation had to deal with other architectural issues like input and output.

Several authors have used simulations or games to teach networking. Voderhobli and Pattison (2005) developed a simulation system for network managers. Guo, Xiang and Wang (2007) developed a simulation network laboratory emphasizing the dynamics of network protocols rather than configuration. Leitner and Cane (2005) designed a virtual networking laboratory intended to address the need for a hands on experience in a distance education environment. Dennis (2002) and Pendegraft (2002, 2003) developed in class games to teach TCP/IP. These two games will be described in more detail in the next section. Leitner and Cane (2005) describe a virtual laboratory for distance education in IS. They argue that simulations do not provide direct experience. They distinguish virtual laboratories from simulations in that the former "is a true laboratory in which experiments are carried out under the control of remote users" (p.284).

There are many interesting efforts to improve the teaching of programming through the use of games. Several researchers have looked at the use of Robocode which uses a game to teach Java. Long (2007) evaluated Robocode as a vehicle for instruction and found that participants had fun and enjoyed the exercise. They mentioned that the exercise was intrinsically interesting because they learned something, but also that they had fun doing so. Bierre, Ventura, Phelps, and Egert (2006) had their students create simulated battle tanks. The assignment ended with a tournament in which the tanks fought against each other. They report that teams who used this environment learned more than those in traditional or Robocode environments. Alice is a visual environment for teaching programming by allowing them to easily create games or simulations (Alice.org). Students learn to think in terms of objects and their behaviors. Kelleher and Pausch (2007) used Alice to inspire middle school girls' interest in learning to program computers. Baker, Navarro, and van der Hoek (2003) describe a card game that they developed to teach software engineering. This approach is not limited on the software side to programming. Lawrence (2004) used a game to teach data structures.

Others have developed games to help students learn about IS management. For example, Jain and Boehm (2006) developed SimVBSE, a game to improve understanding of value-based software engineering. Students act as project managers to learn the fundamentals of software engineering. Sheng, Magnien, Kumaraguru, Acquisti, Cranor, Hong, and Nunge (2007) and Irvine, Thompson and Allen (2005) describe games used to teach about IS security. Curtin, Carpenter and Ritzo (2006) report on games developed to train help desk staff. They also offer a process for creating effective games.

EXAMPLE: USING A GAME TO TEACH TCP / IP

An example may help to clarify the mechanics of gaming and illustrate its utility. Consider two similar games, one designed by Dennis (2002) and one by Pendegraft (2002, 2003), to help teach how TCP/IP works. Both are published elsewhere and so will not be described in detail here. Both

3 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/simulation-games-virtual-environments-education/14090

Related Content

Shared and Distributed Team Cognition and Information Overload: Evidence and Approaches for Team Adaptation

Thomas Ellwartand Conny Herbert Antoni (2017). *Information and Communication Overload in the Digital Age (pp. 223-245).*

www.irma-international.org/chapter/shared-and-distributed-team-cognition-and-information-overload/176573

Detecting Community Structure in Financial Markets Using the Bat Optimization Algorithm

Kirti Aggarwaland Anuja Arora (2022). *International Journal of Information Technology Project Management (pp. 1-21).*

www.irma-international.org/article/detecting-community-structure-in-financial-markets-using-the-bat-optimization-algorithm/313421

A Methodology for Managing Information-Based Risk

Holmes E. Millerand Kurt J. Engemann (1996). *Information Resources Management Journal (pp. 17-24).*

www.irma-international.org/article/methodology-managing-information-based-risk/51022

Matching Facilitator Style and Agenda Structure in Group Support Systems: Effects on Participant Satisfaction and Group Output Quality

Todd J. Hostager, Scott W. Lester, Marilyn Bergmannand Kathryn J. Ready (2003). *Information Resources Management Journal (pp. 56-72).*

www.irma-international.org/article/matching-facilitator-style-agenda-structure/1240

Linguistic Indexing of Images with Database Mediation

Emmanuel Udoh (2009). *Encyclopedia of Information Science and Technology, Second Edition (pp. 2420-2425).*

www.irma-international.org/chapter/linguistic-indexing-images-database-mediation/13923