Serious Instructional Design: ID for Digital Simulations and Games

Katrin Becker, PhD
Adjunct Professor, Simon Fraser University
President, Mink Hollow Media, Ltd.
Canada
becker@minkhollow.ca

Jim Parker, PhD
Professor, University of Calgary
Canada
jparker@ucalgary.ca

Abstract: Digital simulations and games constitute a distinct category of learning intervention that calls for a tailor-made instructional design approach. This article will provide some background for that claim. A brief summary of several well-known I.D. models is presented as well as typical simulation and game design models. These are then combined to form the new synergistic design process model that takes into consideration key features of all three design disciplines, resulting in a new model uniquely adapted for the design of instructional simulations and games. The paper concludes with some lessons learned from first-hand experience designing simulations and games for instructional purposes.

Introduction

Simulations and games share many properties. In fact, depending upon one's point of view, digital games could be defined as a sub-category of computer simulations. A computer simulation is really just an implementation of a model, and this is true whether you are creating a computer simulation or a live-action training simulation. In this case, 'model' has a distinct definition well known in informatics. In computer science, a *model* is a precise description of a system that includes its variables, behaviors, and their interrelationships (Becker & Parker, 2011). Unlike the typical definition used by many in Ed Tech (Gredler, 2004), this definition does not insist that the model be based in reality, just that it be coherent and consistent. That does not mean that a model can be anything however; in order to create a simulation on some sort of computer, one must first create a very precise and unambiguous description of the model you wish to implement.

Simulations and Games Defined

Digital simulations can be roughly grouped into two main types: experimental and experiential and although designing both types is similar in many ways, it is useful to understand the differences as the initial design phases will differ. Experimental simulations seek to answer some question, while experiential simulations attempt to provide an environment with which one or more users can interact. Computer simulations are quite different from what is normally called an *educational simulation*, which typically refers to in-class, live action, or paper-based activities where the learners are given a pre-defined set of roles and then asked to play out or work through some scenario. There is some overlap to be sure, but the complexity introduced when implementing a simulation on the computer results in a matching increase in the complexity of the design process.

In the computer modeling and simulation community, digital games are normally considered to be a subset of simulations (Parker, Becker, & Sawyer, 2008; Tobias & Fletcher, 2007), yet in education circles simulations and games are thought to represent opposite ends of a continuum. This view has persisted in education since 1969 (Ochoa, 1969) who, in turn got some of her ideas from Clark Abt (1966). At that time, the simulations they were talking about were almost exclusively analog (i.e. non-digital). Technology has evolved to the point that many modern simulations and games look like science fiction by 1960's standards, but the view of sims and games from the perspective of education has remained largely unchanged. One of the biggest problems with insisting that games and simulations be placed at opposite ends of a continuum is that it severely limits one's ability to take advantage of what the technology has to offer in terms of design.

Designing Simulations and Games

Design is an applied discipline that depends heavily on content knowledge of what we are designing. It is a complex activity and even though there are common elements to all design endeavors each discipline has distinctions. It is simply not possible to be an expert designer in the general sense. Further, usually when people design things, they are things people want to use or build, at least eventually, and simply knowing how to apply a design model is usually insufficient preparation. The designer must also have experience actually building the thing being designed, or at the very least using it. Becoming skilled at design always requires hands on experience.

Designing a simulation of something is no different, and in some respects, it is actually more complicated than designing the thing itself. Suppose one wanted to design a simulation of a deep ocean submarine. One would have to know enough about how the submarine worked to be able to simulate it effectively, but one would also have to simulate the environment in which it will operate, AND one would have to know how to design a computer program to do all of these things. Not only that, since it is impossible to represent all aspects of the system with complete accuracy, one would also need to assess which aspects of those environments and systems related to the one in question need to be simulated faithfully and which ones don't.

When simulations are used for training and professional development, it adds even more complexity, because now, it is not enough to know that the simulation is correct, it must also be designed in such a way that the target audience gets the intended message.

Designing Instruction

There exists a general recognition in the field of instructional design that no one method can work in all situations. And even those who support the most structured approaches will admit that these are often best suited to use by practitioners new to the field, by providing a support system. Experts who make use of these models often use them as rough guides, rather than prescriptions (Kenny, Zhang, Schwier, & Campbell, 2005). There are well-known models that promote a fairly linear approach to design, such as Gagné's Nine Events of Instruction (Gagné, Briggs, & Wager, 1992), while others suggest more of an iterative approach (Dick, Carey, & Carey, 2001), and still others advocate an agile one (Piskurich, 2000). Many instructional design models have similar elements and the well-known ADDIE template that often forms the basis for these models (Molenda, 2003) still serves as a reasonable common denominator for all. The acronym became popular much later than the process itself (Branson, Rayner, & Cox, 1975) and in spite of being overly simplified, it remains a very popular model in professional training. The five parts of the ADDIE model are outlined below:

- **Analysis**: defining desired outcomes.
- **Design**: determining how desired outcomes are to be accomplished.
- **Development**: establishing requisite system(s) and acquiring needed resources to attain desired outcomes.
- **Implementation**: implementing design and development plans within the real-world environment.
- **Evaluation**: measuring the effectiveness and efficiency of the implemented system and using collected data for improvement in closing gaps between actual and desired outcomes.

These five parts must be evident in the final model if the resultant intervention is to be recognizable as an educational or training solution.

Designing Simulations

While there is some overlap in between the design of instruction and the design of simulations, the latter introduces a significant degree of complexity. In this case the various phases are different enough that some explanation here is necessary and they have been mapped onto the ADDIE model for comparison.

The process is as follows:

1. Conduct Needs Analysis
2. Design
    a. Define Original System (extent and bounds)
    b. Identify Observable Elements (those items for which we can collect data)
3. Development
    a. Collect Data
    b. Create Conceptual Model (the high level design)
4. Implementation
    a. Create Operational Model (the detailed design)
    b. Translate into Simulation
5. Test

The needs analysis is one of the most important steps and no design and development project should be undertaken without a thorough understanding of why you are doing it and what you hope to get out of it. Typically, the outcome of a needs analysis in this arena is the identification of some performance gap. For our purposes it is assumed that it has already been determined that a simulation or game is an appropriate treatment for the gap we are addressing, and so we are not starting at the very beginning.

The original system includes all that can be known about the actual system being modeled whether or not they can be defined or described. Unfortunately, it also includes everything that isn't known. This is one of the chief ways that simulations can be misleading. If the original system is at all complex there is no real way of guaranteeing that some unknown or neglected aspect is actually important to the outcome. That's why it is so important to verify the model.

The observable elements encompass all of the elements that can be observed or experimented with. The observable elements may consist of a single identifiable entity or some number of entities that constitute a coherent subset. One aspect of the initial system description that is very important is to identify the types of data that are actually obtainable. The data will determine what kinds of questions can be explored and what kinds of experiments can be conducted using simulation. The kinds of data important to the development of a viable computer simulation are not always the same as the kinds of data important in the original system.

Data collection is one of the most critical phases in the development of a simulation and deserves specific attention. It is only through the collection of appropriate and accurate data that we can set up a simulation to run in a fashion that properly reflects the original system on which it based.

The conceptual model is the draft description of the various elements that will go into your simulation along with the events that are associated with each. The operational model is the model of the system that is actually implementable. It contains all of the necessary details and omits those that are not of interest. This is the experimenter's image of the original system, and the term comes from lumping together components and simplifying accordingly.

Finally, in the case of simulation design, the testing applies to the simulation rather than the efficacy of the instruction. That part doesn't really come in to the picture until after we have verified the accuracy of the simulation.

Designing Games

By now, the news that a game can be a useful medium for education has reached almost everyone, and it is hard to find a single education or professional development conference that does not include a track on games. We are told that games are motivating (Marsh, 2010) and that games are even good for us (McGonigal, 2011). Hardly a week goes by that one does not find an article on games in newsfeeds devoted to education, formal schooling, and professional or corporate training. However, designing a game involves much more than simply wrapping a game around instruction, and a game cannot be thought of a merely a vehicle for the delivery of instruction. Game design is a highly complex process and many volumes have been written that offer advice on how to proceed. As one example, we will look at the time-tested approach used by Chris Crawford, a game designer perhaps best known for his game *Balance of Power* (Crawford, 1985). In his 1982 book, The Art of Computer Game Design (Crawford), he outlines seven main phases in the design process, a variation of which is presented below:

1. Choosing a goal and a topic (Objective and Premise).
2. Research and preparation.
3. Design Phase
    a. Input Output Structure (Interface)
    b. Game Structure (Gameplay and Game Mechanics)
    c. Program Structure
    d. Evaluation of the Design
4. Pre-Programming Phase
5. Programming Phase
6. Playtesting Phase
7. Post-Mortem

In videogame design, the goal or objective normally refers to the object of the game, in other words, what the player must do to get to the end or win. The premise on the other hand is the means of expressing the objective and may be chosen separately from the objective or they may be interconnected and thus choosing them together is appropriate. Research and preparation in this context are associated both with looking for information about the premise of the game and also with looking for other games related games. In game design it is important to be unique.

Most of the time in a game we think of the interface as being what we can see on the screen, however, the interface also includes how the game is controlled, how information is presented to the player, and the game audio (the sounds and music). In other words this includes anything not directly part of the gameplay.

Game mechanics are the mechanisms by which the player achieves the goals of the game. They include the actions that the player can perform like: collecting, aiming, moving, and choosing. Game mechanics on the other hand can usually be closely tied to the implementation of the algorithms and programs that facilitate the game mechanics. The game mechanics describe what the player can do in the game. Gameplay could include terms like cooperative, leveled, or 2D platform.

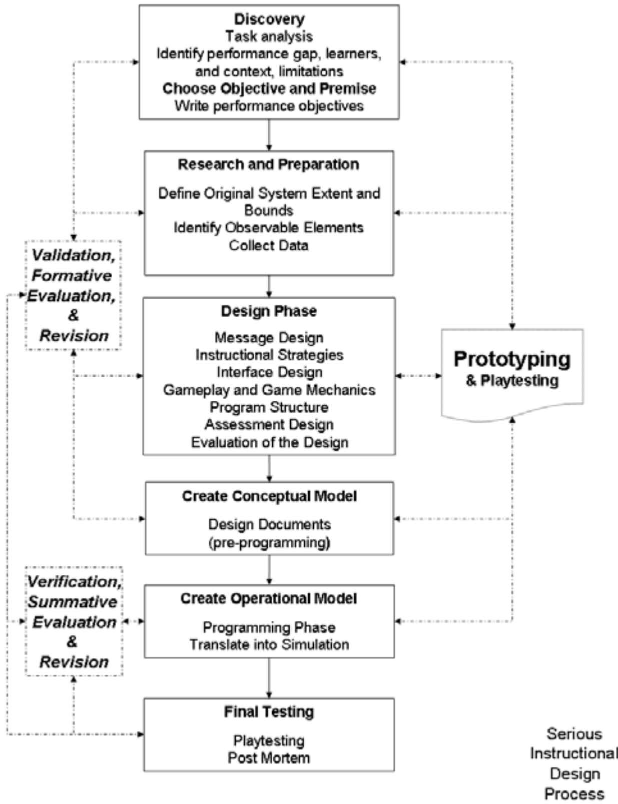
The program structure should be fairly self-explanatory while the design evaluation seeks to expose potential implementation difficulties or inconsistencies that could cause difficulties during the design. The result, once these phases are largely complete, is the “game design document” which follows a fairly well-known format and serves as the ‘bible’ that will be used to guide development of the game itself.

Playtesting involves having people play your game and provide you with feedback on everything from the interface and game mechanics to the dialog and overall appeal. Ideally, the playtesting phase will only uncover minor flaws and misconceptions that will allow the developers to polish and refine the game, but sometimes it reveals major problems.

The final phase of the game design process usually includes a critical examination of the entire process and is typically written up in a document known as the post mortem.

Synergy: Serious Instructional Design

Often, a commercial game design is built up from a single core idea - something (either some activity or some premise) the designer finds amusing or entertaining. Simulations on the other hand are built up to answer a question (or series of questions in a coherent domain), and educational interventions are built up from identified performance gaps. Not only are the different design processes built up from different initial motivations, the traditional or accepted design and development models are also different, and this makes sense. The focus for each of these objects is different: simulations are necessarily concerned with accuracy; games focus on providing an entertaining player experience, and instruction tends to be largely focused on content. So then how do we reconcile and combine these approaches? The short answer is that we don’t, at least not so long as the simulation or game is viewed as the ‘instructional material’ rather than being an integral part of the entire process. What we do is to combine simulation design (S.D.), game design (G.D.) and instructional design (I.D.) into some process that is coherent while remaining simple enough to be useful.



The initial phase of the process is called the 'Discovery' phase in our new model and is meant to encompass all the needs analyses and rough outlines that will be needed to place the remainder of the project in its proper context. As the simulation or game effectively IS the instructional strategy, it is assumed that most of the needs analysis that would normally have indicated the need for a simulation or game will have already been done. You may already know as much as you need to about the characteristics of the learners - their ages, backgrounds, what they already know, and where they are hoping this will lead - but if you don't, the time to find out is *before* you begin to design your solution, so this too is part of the discovery phase.

The research and preparation performed here is largely the same when we are creating an instructional simulation as it is when we are creating one for other reasons, except that the decision over whether or not some aspect of the original system is important will be determined by its relevance to the instructional goals. Research and preparation is a part not normally found in most instructional design models.

The design phase is where the simulation or game will take shape. It is important at this phase to maintain connections between the overarching goals, which are instructional, and the simulation details or gameplay. Although it is not necessary for every aspect of the simulation or game to further the instructional objectives, it is necessary that they coincide often enough to ensure that the time spent in the simulation or game is time well spent.

Creating the conceptual model is another that is not typically found in most instructional design models, but it is one that is essential when designing instructional simulations and games. Since the application will form the primary 'delivery method', this part must be completed carefully. This is effectively the last stage where it will be feasible to back up for major revisions if problems are detected. The outcome of this phase will be the detailed design document, and it should incorporate both the design elements of the simulation or game and the checkpoints needed to ensure that this solution has a reasonable likelihood of delivering on its instructional objectives.

The final two phases have already been described in a previous section and as they are effectively the same as in simulation and games design generally, they will not be explained again here. That leaves the evaluation elements. Formative and summative assessment remain as important in serious instructional design as they are in any other instructional design effort, but if the intervention is destined to be a game or simulation there are the additional requirements of validation and verification. In simulation, validation involves testing the conceptual and sometimes the operational model to make sure that the assumptions and data are reasonable and correct. This part is absolutely crucial to ensuring the accuracy of our simulation. Verification in this context involves testing the code to make sure it functions properly.

Conclusion

This paper progressed through three major design disciplines and concluded with a synergy that combined all three in a serviceable, unified model. The design of instructional simulations and games can be orders of magnitude more complex than designing an entertainment game or a straight simulation intended to produce numerical output to answer a question. The use of a simulation or game for instructional purposes places a different weight on the interaction, as that is the primary interface through which the learner will acquire the desired message or experience. The instructional aspect can neither be an add-on to a game design, nor can the game or simulation design be treated the same way as the 'development and/or selection of instructional materials' is in most common instructional design models. The synergy must be coherent and complete.

References

Abt, C. C. (1966). *Games for Learning*. Cambridge, Mass.

Becker, K., & Parker, J. R. (2011). *The Guide to Simulations and Games*: Wiley Inc.

Branson, R. K., Rayner, G. T., & Cox, J. L. (1975). *Interservice procedures for instructional systems development: Executive summary and model* (Contract Number N-61339-73-C-0150 ed.). Ft. Benning, Georgia: Center for Educational Technology at Florida State University for the U.S. Army Combat Arms Training Board.

Crawford, C. (1982). The Art of Computer Game Design Available from http://www.vancouver.wsu.edu/fac/peabody/game-book/Coverpage.html

Crawford, C. (1985). Balance of Power (pp. PC): Mindscape, Inc.

Dick, W., Carey, L., & Carey, J. O. (2001). *The systematic design of instruction* (5th ed.). New York: Longman.

Gagné, R. M., Briggs, L. J., & Wager, W. W. (1992). *Principles of instructional design* (4th ed.). Fort Worth, Tex.: Harcourt Brace Jovanovich College Publishers.

Gredler, M. E. (2004). Games and Simulations and Their Relationships to Learning. In D. H. Jonassen (Ed.), *Handbook of research on educational communications and technology* (2nd ed.). Mahwah, N.J.: Association for Educational Communications and Technology., Lawrence Erlbaum.

Kenny, R. F., Zhang, Z., Schwier, R. A., & Campbell, K. (2005). A review of what instructional designers do: Questions answered and questions not asked. *Canadian Journal of Learning and Technology, 31*(1), 9-26.

Marsh, T. (2010). Activity-Based Scenario Design, Development and Assessment in Serious Games. In R. V. Eck (Ed.), *Interdisciplinary Models and Tools for Serious Games: Emerging Concepts and Future Directions*. Hershey, PA: IGI Global.

McGonigal, J. (2011). *Reality is broken : why games make us better and how they can change the world*. New York: Penguin Press.

Molenda, M. (2003). In Search of the Elusive ADDIE Model. *Performance Improvement*.

Ochoa, A. (1969). Simulation and Gaming: Simile or Synonym? *Peabody Journal of Education, 47*(2), 104-107.

Parker, J. R., Becker, K., & Sawyer, B. (2008). Re-Reconsidering Research on Learning from Media: Comments on Richard E. Clark's Point of View column on Serious Games. *Educational Technology Magazine, Jan-Feb 2008,* 39-43.

Piskurich, G. M. (2000). *Rapid instructional design : learning ID fast and right*. San Francisco, Calif.: Jossey-Bass.

Tobias, S., & Fletcher, J. D. (2007). What Research Has to Say About Designing Computer Games for Learning. *EDUCATIONAL TECHNOLOGY, September–October 2007,* 20-29.