Defining the Development Pipeline for Meaningful Play

Robert L. Appelman, PhD
Instructional Systems Technology - Indiana University

Keywords: Serious Games & Simulations; Meaningful Play; Production Management; 3D Pipeline Development

Abstract

Meaningful Play refers to the task engagement of a player interacting with a game or simulation designed to meet learning objectives for that player. The *Development Pipeline* refers to the complex series of interactions between both the *Design Team* and the *Development Teams* as they move from the first concept stages to a finished alpha-level 3D virtual game or simulation. This paper introduces the role of the *Instructional Designer* into this pipeline, and also describes how a web site can be used to facilitate the micro-decisions made by each team.

Introduction

Game design, especially that type of game that involves learning as an outcome, are as a whole very undefined in terms of the sequence of decisions that need to be made, who makes these decisions, as well as what development tools are used to make and execute these decisions. I am going to refer to the above as the "*Development Pipeline*" in this paper. Games involving learning as an outcome are often called serious games, but for the sake of embracing more broadly accepted definitions in all fields, I will be avoiding this label, and instead focus on the learner and the ludology of their meaningful play. I do this because it is the learner's execution of appropriate tasks within a game or simulation that will foster their learning, and ultimately for them to form meaning through the process of play. Also, I have been focusing on the production pipeline in game development for the last 10 years, and I believe that I am finally beginning to see some consistencies emerge in the game development process which could be helpful, both from a development design standpoint as well as forming a pedagogical strategy for the teaching of games and simulations involving learning/performance outcomes.

When Clark Abt wrote his book entitled Serious Games in 1970, computers were just beginning to simulate Dungeons and Dragons type of adventure games called MUDs (Multi-User Dungeon). Abt alluded to some computer assisted games such as "Corridor" and "Politico" that had educational value, but primarily he focused on the face-to-face games that were based on inter-personal interactions of groups and the high-level thinking and problem-solving skills that could be an outcome of this activity (Abt, 1970). Although face-to-face games and simulations have been utilized since John Dewey launched his *Experiential Learning* concepts in the early 1900's and later refined by experts such as Thiagi in the present century, the use of virtual environments to accomplish the same levels of learning is only a decade old (Dewey, 1938; Sivasailam Thiagarajan, 1996). Because it is so new, this paper will focus solely on the development pipeline of virtual learning environments, and not face-to-face or board games.

It is not as if there has been nothing written about the development process, because there are many books on game development focused on the organization of teams and the division of labor, and much more on the storyline development (Bates, 1999; Bethke, 2003; Pedersen, 2003; A. a. n. A. Rollings, Ernest, 2000; A. M. Rollings, Dave, 2000; Salen & Zimmerman, 2004). There is a dearth of literature surrounding descriptive models for ways of thinking about design and methodology, but there is a lack of consistency in the literature allowing any one preferred organizational or procedural strategy to surface (R. Appelman, 2005; Crawford, 1984; Gee, 2003a; Herz, 1997; Klabbers, 2000; Klassen, Vogel, & Moody; Prensky, 2001; Squire, 2003; Swartout, 2003; S. Thiagarajan & Thiagarajan, 2001; Vandeventer & White, 2002). Klabbers points out that there is a tension created by the fact that multiple disciplines with differing points of view must be thrust together in this development environment, and this supports my supposition that the way things are done is more correlated with the unique group make up within each company, or set of individuals, who are designing the game. This diversity has prevented the game industry from allowing common strategies and models for the game design & development to arise (Klabbers, 2003). This paper is an attempt to bridge this dialog in such a manner as to foster the identification of key decision areas that are common in the development of any immersive virtual learning environment.

Play vs. Learning & Games vs. Sims

In a game/sim designed for learning, everything begins with the establishment of learning objectives for a particular group of learners. Each group of learners will have a particular unique set of demographics and will encounter this game in a specific context. There is no way to define a general context , because there is too close a tie of any context to student motivations and interests towards the subject matter, let alone their willingness to even be involved with a game-like pedagogy (R. Appelman & Wilson, 2006; Hirumi, Appelman, Rieber, & Van Eck, in Review; Kirkley, Sonny E. Kirkley, Myers, Lindsay, & Michael J. Singer, 2003; Salen & Zimmerman, 2004). An example to illustrate this might be when a student is on a playground swinging in a swing, the student's mental focus is not on analyzing pendulum motion, but just on the experience of swinging. Attempting to use this context for learning by interjecting inquiry about how they were swinging would be a distraction to the student and not very productive. However, should you bring a swing into a classroom as a model to analyze, the state of mind of the student changes and is focused on learning, thus the focus on the content of pendulum motion, coupled with the enjoyment of using such a well known tool of fun, could make the learning very pronounced.

It is necessary at this point to delineate the difference between a game and a simulation. At the most basic level, a game is based on a series of rules or algorithms that operate within the game, and that the player must manipulate and abide by to achieve a specific end to the game. In a simulation the player must make decisions that are based on an underlying model within the game to achieve various affects and consequences within the game. It is the analysis of one's decisions relative to their consequences that allow for high-level learning and thinking skills to form as a result of meaningful play. In a simulation this reflection can occur within the game or outside the game; especially if an instructor or facilitator creates an appropriate scaffolding outside of the game that enhances the scaffolding existing within the game. Given these two distinctions, a game based on a set of known relationships or rules is ideal for becoming familiar with, or practicing, a specific set of content. Game simulations, on the other hand, are excellent for ill-defined problem-solving, and dealing with content

that could have multiple solutions, multiple consequences, and for which there may not be a right or wrong answer. With a simulation, an evaluation by the player that some results could be better than others in different contexts is often the goal of learning. The majority of virtual games produced to date have been based on algorithmic rule-based engines and are primarily designed to give a student drill and practice exercise. The majority of face-to-face, or group activity games, have been more aligned with game simulations, in that they addressed more complex decision-making (R. L. Appelman, 2007; Duke, 1974; Gee, 2003a; Prensky, 2001; Salen & Zimmerman, 2004; Sellers, 2002; Summers, 2004). In this experience the decisions made by the player will be discussed and weighed both within the game and also outside of the game. It is important to note that a strong motivation or for using a game, is to create a context that is nonthreatening to the player, yet generalizable to more real contexts outside of the game experience. It is also possible to embed many games within a simulation. This is sometimes done for remedial purposes if the game engine and artificial intelligence of the game is smart enough to recognize that the player needs remedial assistance by noting deficiencies of game play .

Virtual Game Development

The following developmental process describes a pipeline that targets a virtual computer-based game or simulation. This game could be released on a console game, a PC, or a number of other handheld or telephone devices. For the purposes of this paper a specific game designed to teach physics and math concepts will be used for illustration called iKids© . Specific reference to this game design will follow the developmental descriptions and preceded by the word *"example:".*

Four Teams

There are four primary teams that need to be managed within the scope of the game development. These consist of the *Game Design Team*, the *Programming Team*, the *Art Team*, and the *Audio Team*. There needs to be one or more primary leaders within each of these teams, and depending upon the scope of the game or simulation, there will need to be specific attention focused on the communication between and among all group members. Because of the unique dynamic development process of computer 2-D and 3-D design, the communication is more critical in this development pipeline than in the motion picture development process. The reason for this is due to the greater creative contribution that each interdisciplinary team member has to the final product. Correspondingly, it is necessary to use very efficient communication and organizational tools to facilitate this process so that everyone know just what each team is doing; however, the tools and documents produced for game development are as varied as the number of game companies and groups that produced them. In contrast to the rigid set of documents produced for a film (Scripts, Screenplays, Storyboards, etc.), the traditional document produced to guide the development of a game is called the *Design Document*. In this document the general strategies, general functionality, and the general game play the player will encounter is defined, however it cannot be too specific prior to full character and functionality development. Both Rollins and Bethke describe design documents that are very extensive consisting of hundreds of pages, describing not only game play, but specific art forms, programming architecture, and audio style, but it

is also assumed that there will be many changes and variances from the *Design Document* and it is often ignored or abandoned because it is either not accurate or not current. The divergent nodes a player will encounter, and the resulting issues of tracking branching Boolean states, will cause variances of not only style, but functionality and content. The four interdisciplinary teams must be able to follow micro-decisions made by each of the other teams that may have consequences on their own team decisions.

The Game Design Team

The *Game Design Team* is made up of an *Instructional Designer*, a *Creative Designer*, and a *Project Manager*. The development pipeline begins with the determination of learning objectives under the lead of the *Instructional Designer*. At the very beginning specific content is identified that is tied to what a student is ideally going to learn with this game/sim. Also, since learning itself is relative to what the student already knows about that specific content, the student's entry-level with the content is also defined. Once this relationship of *what is desired to be known* and *what is already known* is quantified, another aspect of the instructional design equation – the *level of motivation* the student has for gaining this new knowledge is added. It is in the description of the motivation that the degree of "fun" the game must embody can be defined, as well as how difficult the meaningful play will be to acquire this new knowledge.

- If a student has little knowledgeable about the subject matter, but has a high level of motivation to learn it, then the degree of difficulty of the reward system and of the path to discover this knowledge can be made difficult, but with the caveat that there must be a safety net of coaching help and content resources available.

- With the student that is highly motivated and also highly knowledgeable of the subject matter, one can offer a complex set of tasks in meaningful play, and a complex nodal structure that would have little coaching. The caveat here is that the learning context or problem must be very authentic to make it engaging for the learner.

- A moderately motivated student with moderate content knowledge would require an equally moderate level of complexity, coaching, and highly affective engaging components. The theory base for these examples resides with James Gee, Piaget, and also a series of combinations linking constructivism and experiential learning ala John Dewey in the 1930s (Dewey, 1938; Gee, 2003b; Jonassen, 1999; Piaget, 1950; Sivasailam Thiagarajan, 1994).

example:

The Player: Entry Level is 8^{th} – 9^{th} grade Algebra, Trigonometry, & Basic Physics. Motivation is moderate for Math, but high for Physics.

The Learning Objectives: within the game/sim the player will encounter problems, and successfully demonstrate ability to:

1. Identify trigonometry formulas dealing with sides and angles of a triangle
2. Recognize contextual variables that must be measured and/or manipulated to solve specific problems
3. Identify and manipulate tools that measure angles and distances
4. Etc…

The Context for the learning must be situated in authentic contexts where physics problems are encountered. The perception of the player must be that s/he is among a group of peers, each with something to contribute to the problem, and may be operable in both single and multi-player modes. Collaboration among players or interaction with NPCs and/or other resources will be necessary to solve problems.

Once this not so trivial task of defining the learning objectives, the audience, and the context is defined, then the *Creative Design* begins.

The Creative Design

*Creative Designer*s often start with a story that describes the content and how that content will unfold in the specific context of game play. This context may be fantasy, it may be realistic, or it may be a combination of the two. It will necessarily require that whatever the context, it must allow the player to be engaged with specific content as defined by the *Instructional Designer*. The difference in focus with *Meaningful Play* is that the *Instructional Designer* is specifying content and tying it to learning acquisition of the player, while the *Creative Designer* is focused on the form of the play environment and such things as aesthetics, continuity, pacing, and player experience. The process between these two designers is one of collaborative design spiraling around a constant give and take of form and function. The basic design element that the design group uses to have the player encounter and engage in content is the task. Each player will encounter various tasks to achieve in order to reach the level goal, and these levels are used to stratify the content into different areas and difficulties.

The Case Scenario

This *Case Scenario* is the first item to be delineated and should appear on the development web site. This development web site should be available to all team members and will track all decisions made as the game/sim is developed. Similar to a story that unfolds the characters, context, and specific interactions, it is possible to identify specific locations where this action occurs. We will call these locations *Scenes,* and the general sequence of decision-making that the protagonist (who in our game/simulation is the *Player*) would need to make will be called *Tasks*. Applying this terminology one can say that, the *Case Scenario* will be described such that specific *Tasks* are identified in each *Scene* that the *Player* will need to address. The *Consequences* of a *Player Action* (choosing to make a decision or not, or choosing one of multiple options available) is first delineated in terms of an original case study. In other words, the *Game Play* is laid out in terms of there being only one choice for each *Task*. Following this strategy of structuring the game/simulation directly following the original case study underlies the basic goal of maintaining authenticity in this "rapid prototyping" development scenario. Once the original case study decision nodes are identified, then the possible options and resulting consequences can be built on top of an authentic foundation.

example:

The Context of the game/sim will be in the future, with a group of teenagers called the "Incredible Kids – or iKids©" who have exceptional skills in math and physics. Although these iKids© operate in the future, the majority of game play and problem solving occurs when they "beam back" to the past where they interact with historical characters who developed some of the scientific theories we use today. A portion of a script (Jointly produced by both the Instructional and *Creative Designer*s) follows:

INFO Elements	INTERACTIONS	STORY
o DATE o LOCATION (and maybe	o Manipulation of the HISTORY SCANNER o Comments to each other	The Incredible Kids are hanging out in their special lab-lounge and a signal is broadcast to

in a Google Earth fashion) o IMAGE/AUDIO of LEONARDO as he mutters to himself about the concept he is wrestling with o PRINCIPLE ANALYZER gives readout of key concepts of math, physics, and tools they need to take with them to assist.	through NPC dialog o Manipulation of PRINCIPLE ANALYZER	them from their HISTORY SCANNER. They run over to this to find out where the need is this time to "help history" continue on its projected path. This time the signal is coming from the 16th century and specifically from Leonardo DaVinci's Lab in Italy. The scanner has picked up a "Concentration Center" where Leonardo is working on a concept of gravity, but seems to be having problems. THIS IS THE TIME FOR THE iKIDS TO BEAM THEMSELVES THERE TO HELP.
o Costumes of each of our iKIDS automatically change period.	o PC must locate and begin interaction with LEONARDO	The iKIDS are transported in, of course a very neat fashion to Italy, and specifically into Leonardo's Lab
--JUMP IN SCRIPT--	*--JUMP IN SCRIPT--*	*--JUMP IN SCRIPT--*
o FORMULA o VARIABLES o Defining a CONTEXT for INQUIRY that matches the problem o BLENDING MATH and PHYSICS to solve a problem	o Looking at resources o Turning pages o Pointing at critical elements through cursor roll-overs where key elements highlight o MAPs of ITALY that show location of PIZZA	At this point our Physics NPC can look at one of the resources he brought and look up "WHEN THINGS DROP". S/he comes across the formula for calculating the speed of a falling object that has distance, & mass times a constant. The Math NPC then says oh that's easy and says all we need is a known height of something and drop two different mass objects. Leonardo says that he knows the perfect place for this … the Leaning Tower of Pizza. One of the other NPC's says "It's made of PIZZA?" and Leonardo, after laughter stops, says it's a town.
o Structural Engineering Content	o Navigating around tower	The group goes to the town of PIZZA and navigate to the Tower. One of our NPC's asks why it is leaning and Leonardo explains that they did not test the ground adequately under the structure because they thought it was stone. Leonardo could also quip that he "told them so, but they would not listen to him".
--JUMP IN SCRIPT--	*--JUMP IN SCRIPT--*	*--JUMP IN SCRIPT—*
o TOOLS THAT MEASURE DISTANCE o TOOLS THAT MEASURE ANGLES	o Locating tools in backpack	Our PC asks how tall it is, and Leonardo says that he does not know that. At that point our PC says don't worry, that's what we are here for, and then looks in his backpack for two things … a pedometer and a sextant, which are labled "HOW FAR DO I WALK" and "WHAT IS THE ANGLE".
o TRIGONOMETRY FORMULA FOR SIDES AND ANGLES OF TRIANGEL	o Find Formula in Math Reference	The Math NPC mentions s/he has the formula for determining height and shows it to everyone
o HOW A PEDOMETER WORKS o HOW A SEXTANT WORKS	o ENTERING PACE INTO PEDOMETER o Navigating to base of Tower and then taking measured steps back to group o SIGHTING THROUGH SEXTANT and reading output	Our PC walks to the base of the tower and sets the PEDOMETER to his STRIDE distance, and walks back to the group. He then asks his Math NPC to enter the result displayed on the PEDOMETER into the correct location in the formula. Then our PC takes the SEXTANT and sights up the top of the tower and reads the angle which is given to the Math NPC.

Scene Definitions

Once Scenes are defined (such as the ikids lab, Leonardo's workshop, etc), the character that represents the player (one of the iKids), other characters (the other iKids, Leonardo, etc.), or objects that our player will encounter or confront (the History Scanner, their Backpaks, their Resources, etc.) can be defined. These other characters will be called *NPCs* (Non-Player Characters), and the *Objects* are items within a *Scene* that may be interacted with by the *Player*. *NPCs* and *Objects* are all tracked by the *Scene(s)* in which they appear. This is where the web site becomes critically important because interaction with these elements will often vary depending upon previous interactions. If a *Player* interacts with an *NPC* multiple times within a *Scene*, the authenticity and believability of the *Game Play* would be challenged if the response from the *NPC* is identical for each encounter. Thus it will require micro decisions derived from the *NPC's* character definition to provide appropriate and authentic responses for each anticipated encounter.

The Project Managment

The *Project Manager* is in charge of web site development and maintenance that will be the primary tool to facilitate assignments and communication flow among all personnel. Quality Assurance (QA) is also a responsibility that falls under the *Project Manager*, and this task compares the contracted work specifications with what is actually produced, and often requires usability testing to assure appropriate levels of functionality. The goal is to have a web site that is robust enough such that the *Design Team* can provide comments and guidance on asset development in progress such that the QA task is minimal.
Besides tracking assets this web site can provide developmental information such as which Team is working on specific assets of the game. It can show Gantt charts of overall development and milestone settings and it can trigger the need for special meetings to unravel problems that occur.
Placing priorities, determining changes in direction, setting deadlines, and assuming a "Quality Assurance" role is something that only the *Project Manager* can provide; however, a tool such as a development web site will allow all teams to self-manage their own development processes, thus taking a great burden off of the *Project Manager* and creating an esprit de corps that fosters a positive working environment. Below is an example of the top level of a game web site where all underlined items link to specific information that will reflect development status.

example:

iKids© Scenes

NAME	CODE	DESCRIPTIONS by ACTION AREAS
iKids© Lab	IKL	(Beginning of new game) Lounge Area
iKids© Lab	IKL	Resource Area
iKids© Lab	IKL	Control Room Area
Leonardo's Lab	LLA	Atrium
Leonardo's Lab	LLW	Workshop
(continues for all Scenes)		

iKids© Player States

CODE	DESCRIPTIONS of STATES
A	New Game and no Character Selection
B	Character Selection and no familiarity with iKids© Lab
C	Familiarity with iKids© Lab but no interactions with fellow iKids©
D	Familiarity with iKids© Lab and interactions with fellow iKids©
E	Familiarity with Resources

F	Jumped in time to Scene X
	(Each SCENE has separate player states plus some global learning states)

iKids© NPCs

NAME	CODE	DESCRIPTIONS by STATE	AUDIO
James	JMX	IKL: A1 A2 A3 A4 B1 B2 B3 B4 C1 C2 D1 D2 E1 E2 - - LLA: - - F1 F2 - - -	Yes
Maria	MAX	IKL: A1 A2 A3 A4 B1 B2 B3 B4 C1 C2 D1 D2 E1 E2 - - LLA: - - F1 F2 - - -	Yes
Rajat	RAX	IKL: A1 A2 A3 A4 B1 B2 B3 B4 C1 C2 D1 D2 E1 E2 - - LLA: - - F1 F2 - - -	Yes
Leonardo	LEO	LLA: LL1 LL2 LL3 LL4 LD1 LD2 LD3 - - -	Yes
LEO-Servant	LSV	LLW: LL1 LL2 LL3 LL4 LD1 LD2 LD3 - - -	Yes

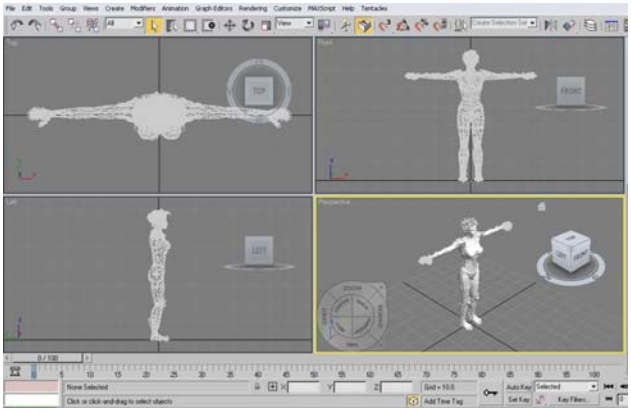
iKids© Objects

NAME	CODE	DESCRIPTION by SCENE & STATE	AUDIO
History Scanner	HSC	IKL only: OFF ON ALERT SEARCH DETAILS - - -	Yes
Principle Analyzer	PAZ	(same in all scenes:) MOTION WEIGHT TRAJECTORY MEASUREMENT - - -	Yes
Sextant	SXT	(same in all scenes:) MOVING PARTS DETAIL VIEWS SIGHTING VIEW DATA	No
(continues for all Objects)			

The Art & Audio Teams

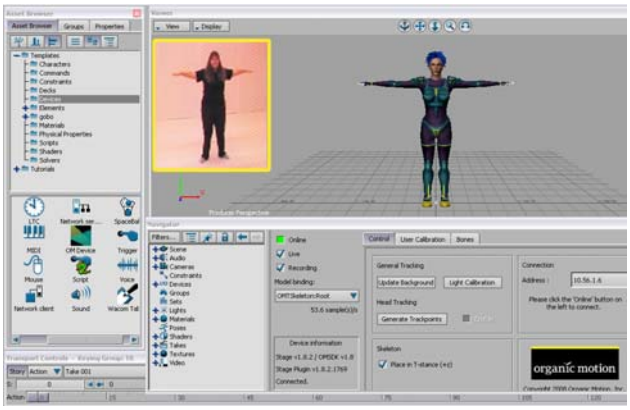
The Art and *Audio Team*s create all that is perceived within the game/sim world. Based on the descriptions laid out in the storyline by the *Creative Designer*, and on the qualities specified by the *Instructional Designer*, the *Art Team* will create specific 3D models for each object or character created software packages such as 3D Studio Max, Poser, and a host of others (Fig 1). The buildings, foliage, and even the clouds in the sky may also be 3D Models built with these same tools. Based on action described in the scripts, the *Art Team* will create animation frame-by-frame, or use another resource such as a VICON's (uses markers) or Organic Motion's (markerless) Motion Capture systems (Fig 2). Both the 3D Model and the data from MoCap are considered assets that the *Programming Team* must draw upon from the asset library and port into World Building software such as the Unity, Unreal, or Vicious Engine (Fig 3).

Figure 1



Model in 3D Max Modeling Software

Figure 2



Model In Motion Builder (Motion Capture) Software (with actor insert)

Figure 3

Model in Vicious Engine (World Building) Software

The *Audio Team*s spends much of their time perusing the descriptions and assets the *Art Team* has created to harmonize sounds to go with them. Each character must have an appropriate voice and delivery of lines. Each object must have appropriate sounds as they are used, manipulated, or simply set down on a table. As characters walk across different terrain their steps will sound different, so there is constant review by the *Audio Team* for everything that is created or specified by the other teams such that they may create the appropriate narrations, sound effects, music, and ambiance to match.

The Programming Team

The *Programming Team* is not only responsible for assembling all of the assets created by the *Art & Audio Teams* into the game/simulation *Map* (the Game Play space/world), but they must also deal with making this world believably functional at a micro level. This includes concerns of "collision detection" (specifying how close another object can get to a target object before it is stopped), employing a variety of "engines" that control movement, physics, gravity, atmosphere, and a host of other functionality. Specific functions involving player control actions and the animation of parts of objects such as tools, doors, picking up things, and a host of other functionality, create a long list of micro-tasks. It is critical for the other teams to place needed functionality on the web site for each character and object so that the *Programming Team* will know what functionality needs to be employed.

Possibly the most important functionality feature of a game/simulation is the *AI* (the Artificial Intelligence) that underlies the programming architecture. The *AI* controls things like "state changes" for each character and object that change before and after a player comes in contact with them. The state change of a character was mentioned earlier to control a different response from an *NPC* whenever a player encounters it multiple times within a game/simulation. To a programmer, these state changes are controlled through "Boolean variables" and is transparent to the Player, because they only experience one path through a game, unless the game allows them to stop and go back and try a different path.

Interdependency of all Teams

All teams must do their part to make the total experience of the player meaningful, effective, and

engaging. The *Programming Team* must be able to offer reality checks for what is possible functionality within any Scene, e.g. the Sextant must work easily and predictably. The affective component of the game is heavily driven by the *Creative Designer*, e.g. delivering a believable story line that is coupled with the creativity from the Art & *Audio Team*s who deliver believable looking 3D models that look and sound appropriate within that story. The *Instructional Designer* monitors the values of each task in terms of meaningful play, and through the collaborative design of all teams, the resulting gestalt is a virtual environment that is not only visually stimulating' but authentic in terms of the content displayed and also an effective learning environment.

example:

Below is one link from the iKids© NPCs table above (James: IKL:C1).
Note how all teams have input into different parts of this web page.

CHARACTER MODEL	3D biped	MoCap Data	James is a white Anglo-Saxon 16 year old who is particularly good with Mathematics. He is somewhat introverted and quiet, but has a dry humor and wit.		
OBJECTNAME	**James [JMX] functioning as NPC**				
PLAYER STATE		**C**	Familiarity with iKids© Lab but no interactions with fellow iKids©		
PLAYER STATES		**C.1**	**C.2**	**C.3**	**C.4**
SCENE:	iKids© Lab Lounge				IKL-Lounge
Initial ACTION:	JMX raises head (or turns head) smiles and says:				
EVENT ID	**PLAYER's POSSIBLE RESPONSES**	**SCRIPT of OBJECT's RESPONSES**			**SOUND FILE**
C.1a	*(upon 1st collision detection)*	Hey there [MAX or RAX], you look pretty relaxed in our lounge here, but have you checked out the Resource Area or Control Room yet?			C1aJMX-MAX.aif C1aJMX-RAX.aif
C.1b	Where is the Resource Area?	Through that door to your right [JMX points to SE corner of Lounge]			C1bJMX.aif
C.1c	What can you do here in the lounge?	Well, besides just hanging out and talking, there are some History Magazines, and a small media library over there [JMX points to N wall of bookcases]			C1cJMX.aif
C.1d	Where is the Control Room?	Through that door to your left [JMX points to NW corner of Lounge]			C1dJMX.aif
Closing ACTION:	JMX looks away to what he was doing				

A programmer would look at the page above and know that s/he could click on the links that the *Audio Team* placed in the "Sound Files" column to hear the actual audio of James speaking his lines in the script. S/he can also access the "3D Model" link to retrieve the actual 3D Max file of the fully textured model of James in the appropriate wardrobe for this scene that was placed there by the *Art Team*. There is access to all of the animation data for this character under the "MoCap Data" link for James in:

- Idle state (for when he is not interacting with anyone)
- Raising Head
- Smiling
- Mouth movements to sync with each audio file
- Gesture to the right
- Gesture to the left
- Lowering Head

The model and animation codes must be linked together using the "World Building" tool, and the 3D Model of the iKids© Lab is also found via the IKL-Lounge link and all of this data must be ported into the world by the *Programming Team*.

Summary

The key concept to make meaningful play effective is collaborative design. To truly embrace a collaborative design approach is to allow oversight and input from all teams, and with the recognition that each team has a specific responsibility to assure effectiveness following standards of their particular discipline. The *Design Team* will necessarily be making compromises based on the "realities" presented to them by the *Development Teams* of Art, Audio, and Programming. The *Development Teams* will need to accept "reconfigurations" of work done to meet standards of the *Design Team*. The singular outcome that gives all the motivation to keep working on their respective teams is the vision of a product that will truly allow a player to have an engaging and learning experience through meaningful play.

List of References

Abt, C. C. (1970). *Serious games*. Canada: Macmillan Company.

Appelman, R. (2005). Experiential modes: a common ground for serious game designers. *International Journal of Continuing Engineering Education and Life-long Learning, 15*(3-6), 240-251.

Appelman, R., & Wilson, J. (2006). Games and Simulations for Training: From Group Activities to Virtual Reality. In J. Pershing (Ed.), *Handbook of Human Performance Technology*. San Francisco: Pfeiffer.

Appelman, R. L. (2007). *Experiential Modes of Game Play.* Paper presented at the Digital Games Research Assoc. (DiGRA), Tokyo Japan.

Bates, B. (1999). *Game Design: The art and business of creating games*. Roseville, CA: Prima Tech.

Bethke, E. (2003). *Game Development & Production*. Plano, TX: Wordware Publishing, Inc.

Crawford, C. (1984). *The Art of Computer Game Design*. Los Angeles: Osborne/McGraw-Hill.

Dewey, J. (1938). *Experience and Education*. New York: Touchstone: Simon & Schuster.

Duke, D. (1974). *Gaming: the futures language*. Beverly Hills: Sage Publications.

Gee, J. P. (2003a). *What video games teach us about learning and literacy*. New York: Paulgrave Macmillan.

Gee, J. P. (2003b). *What video games teach us about learning and literacy*. New York: Paulgrave Macmillan.

Herz, J. C. (1997). *Joystick Nation: How Videogames Ate Our Quarters, Won Our Hearts, and Rewired Our Minds*. Toronto: Canada: Little, Brown & Company.

Hirumi, A., Appelman, R., Rieber, L. P., & Van Eck, R. (in Review). Preparing ID for Game Based Learning. *TechTrends*.

Jonassen, D. (1999). Designing constructivist learning environments. In C. Reigeluth (Ed.), *Instructional-design theories and models* (Vol. II). Hillsdale, N.J.: Erlbaum.

Kirkley, J. R., Sonny E. Kirkley, P. D., Myers, T. E., Lindsay, N., & Michael J. Singer, P. D. (2003). *Problem-based embedded training: an instructional methodology for*

embedded training using mixed and virtual reality technologies. Paper presented at the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2003, Orlando, FL.

Klabbers, J. H. G. (2000). Learning as Acquisition and Learning as Interaction. *Simulation & Gaming, 31*(3), 380-406(327).

Klabbers, J. H. G. (2003). Simulation and Gaming:Introduction to the Art and Science of Design. *Simulation & Gaming, 34*(4), 488-494.

Klassen, J., Vogel, D., & Moody, E. Designing for Interactivity.

Pedersen, R. F. (2003). *Game Design Foundations*. Plano TX: Wordware Publishing, Inc.

Piaget, J. (1950). *The Psychology of Intelligence*. London: Routledge and Kegan Paul.

Prensky, M. (2001). *Digital Game-Based Learning*. New York: McGraw-Hill.

Rollings, A. a. n. A., Ernest. (2000). *Andrew Rollings and Ernest Adams on Game Design*. Boston: New Riders.

Rollings, A. M., Dave. (2000). *Game Architecture and Design*. Scottsdale, AZ: Coriolis.

Salen, K., & Zimmerman, E. (2004). *Rules of Play*. Harvard, MA: MIT Press.

Sellers, M. (2002, September 16, 2002). *Creating Effective Groups and Group Roles in Massive Multiplayer Games.* Paper presented at the Gamasutra, San Jose, Ca.

Squire, K. B., Sasha. (2003). *Engaging urban underserved students in learning world history through computer simulation games*.Unpublished manuscript.

Summers, G. J. (2004). Today's Business Simulation Industry. *Simulation & Gaming, 35*(2), 208-241(234).

Swartout, W. R. v. L., Michael. (2003). Making a game of system design. *Communications of the ACM, 46*(number seven).

Thiagarajan, S. (1994). How I Designed a Game - And Discovered the Meaning of Life. *Simulation and Gaming, Silver Anniversary Issue*(Part 2), 529-537.

Thiagarajan, S. (1996). Instructional games, simulations, and role-plays. In R. L. Craig (Ed.), *The ASTD Training and Development Handbook* (4th ed., pp. 517-533). New York: McGraw-Hill.

Thiagarajan, S., & Thiagarajan, R. (2001). Puzzle Pieces. *Simulation & Gaming, 32*(2), 263-265(263).

Vandeventer, S. S., & White, J. A. (2002). Expert behavior in children's video game play. *Simulation & Gaming, 33*(1), 28-48(21).