



**DEEP LEARNING PROPERTIES OF GOOD VIDEO GAMES:
HOW FAR CAN THEY GO?**

James Paul Gee
Mary Lou Fulton Presidential Professor of Literacy Studies
Arizona State University

In earlier work) I argued that good commercial video games provide players with “good learning” (see Gee, 2003, 2005, 2007). By “good learning” I mean learning that is guided by and organized by principles empirically confirmed by systematic research on effective and deep learning in the Learning Sciences (Bransford, Brown, & Cocking 2000; Gee 2004; Sawyer 2006). This actually should not to be surprising. Video games are, at their heart, problem-solving spaces that use continual learning, and provide pathways to mastery through entertainment and pleasure. Not surprisingly, there has been a growing interest recently in so-called “serious games;” games that involve learning the sorts of domains, skills, or content that we associate with school, work, health, citizenship, knowledge construction, or community building, and not limited to “pure” popular form of entertainment (i.e., Witchcraft, sorcery, fantasy war, etc.).

Games can be used for different types of learning. For example, we could (and do) use games for skill-and-drill, for a sort of trivial pursuit that takes knowledge to be memorizing and repeating “facts.” Or we could seek to use games for the creation of deeper conceptual understandings and for problem-solving abilities that go beyond being able to pass paper-and-pencil tests. The creation of deep “serious games” for such deep learning remains today more a hope for the future than a realized possibility, though there are intriguing beginnings here and there.

There are lots of features of good commercial games that make them good for learning. The most obvious learning is simply how to learn to play the game by learning rules, procedures and causes and effects. A deep “serious game” would, of course, be a game we wanted people to learn how to play because we believed that learning to play it would involve content, skills, values, and conceptual understandings that we believe are

important—a game, for example, devoted to urban planning, social activism, some type of science or business, or the exemplification of a particular perspective connected to ways of understanding and changing the world. There are, of course, examples of such games already available. How well they compare to commercial games and in what ways is an open question.

What I want to do in this paper is discuss what I think are the deepest and most important properties of commercial video games that allow them to achieve powerful learning effects, in the sense both of learning to play the game (and the content and skills thereby involved) and of creating commitment and attachment to play and learning in the game. I would argue that if we are to make deep “serious games” that really use the power of gaming, then these features will have to be present and implemented well. In the end, I am not sure this can always be the case when we leave the domains (“content”) usually covered in commercial games, though this is a matter for future research. That it can be done in some domains is certainly suggested by the fact that it has already been done to a certain extent in commercial games like *Civilization* or *SimCity*, games that connect to domains (e.g., history, geography, urban planning) that we think of as “serious.” How far this paradigm can be extended is, again, an open question.

Property 1: “Gaming” as Psyching Out How Rules can be Used for One’s Advantage to Accomplish Goals to which One is Personally and Emotionally Attached

Consider the phrase “gaming the system”, which means: using the rules or policies of a system or institution against itself. “Gaming the system,” in this sense, is,

oddly enough, close to the core meaning of “gaming” in the sense of playing video games. At its foundation, (video) gaming is about discovering how **the rules of a game can be used to a player’s advantage in order to accomplish the player’s goals.**

When I use the term “rules” here I mean both rules that the game’s designers have put in the game and rule-like properties players’ discover and exploit (“emergent properties”). In this sense gaming is always about problem solving. But, crucially, this problem solving is integrated with self interest. There is something **personal at stake for the player in solving the problems.** It is personal, not just in the sense of “winning” or “losing” (which is not required), but in the sense of accomplishing goals to which the player is personally and emotional committed **in the way in which people are often personally and emotionally committed to winning and losing.**

It might be objected at this point that the goals in a video game are set by the designers, not the players. Therefore, they are not really, or, at least, not deeply the player’s goals. But this is not, in my experience, how gamers look at game goals. First, players accept game goals via the act of having chosen to play the game. Second, they adapt and transform the goals personally by seeking to accomplish them in their own way (style, skill level, their own standards of accomplishment—thus, some players replay bosses they have beat, or repeat whole levels of games, to “do it better”). Good games allow for this and often offer multiple ways to solve a problem. Third, in many games, gamers can set goals of their own, which they must, of course, accomplish within the parameters afforded by the game’s rules (e.g., getting through the museum level of *Thief: Deadly Shadows* without killing any guard or ever being seen). Fourth, in some games (e.g., *The Sims*) players set almost all the goals themselves.

So gamers **care** about their goals in a game and problem solving is personal. We know from considerable research on the human brain that people learn more deeply when there is an emotional attachment to their learning and problem solving, when something is “at stake” for them personally (Damasio, 1994, 1999, 2003). Fear is often the effective emotion for learning in life, but it does not have to be—other emotions work as well. Saying there is an emotional attachment means that **something is felt to be personally “at stake” for the player or learner in problem solving**. Winning and losing is one way—though not the only way—this effect is created in games.

Property 2: Gaming as Micro-Control that Gives Rise to Either

Embodied Intimacy or a Reach of Power and Vision

Many but not all games have avatars which the player controls—like the master thief Garrett in the *Thief* games or Solid Snake in the *Metal Gear Solid* games. However, in video games in general the player “micro-controls” one or more elements in the game. By “micro-control” I mean that the player can affect the movements and actions of that element or elements at a fine-grained, detailed level.

Of course, micro-control is readily apparent when we manipulate Lara Croft in the *Tomb Raider* games or the Prince of Persia in the *Prince of Persia* games. However, in *SWAT 4* the player manipulates one policeman and gives orders to three others; in *Full Spectrum Warrior* the player does not directly move any one character, but gives orders to two (sometimes three) squads of four soldiers each. In *Rise of Nations* or *Age of Empires* the player builds and manipulates all sorts of elements: soldiers, workers, units, buildings, vehicles, and monuments. In *Tetris*, the player manipulates the whole set of play pieces, being able to twist and turn each one as it falls.

Research on learning and the brain has discovered that such micro-control has an interesting and important effect on humans (Clark, 1997). The space over which humans feel they have direct and immediate micro-control is the space within which we humans feel we have “embodied power.” It is the space which we feel, in some sense, our body fills. This space has been, for most of human history, the space intimately close to the human body, just the area we can touch and feel. Blind people learn to extend this sense of the space they directly control out to the tip of their cane, thereby extending it. With the introduction of the Internet, strange effects can be achieved: if a person is using a web cam to manipulate a watering can in a fine-grained way to water plants in another country, the person feels that his or her body has, in a sense, extended to that other country.

Video games, in giving players micro-control over an element or elements in a virtual world, create an effect where the player feels that his or her body has extended into and is intimately involved with the virtual world. This creates an effect that all gamers are aware of: a type of melding of one’s self and one’s character (especially with agile avatars like the Prince of Persia, the God of War, Lara Croft, Garrett, or Solid Snake), a melding of one’s real body and one’s surrogate body. What this means is that cognition (thinking and problem solving) in a game are “embodied”—and the research evidence shows that we humans learn best when we think and problem solve through experiences we are having as embodied beings in the world.

Games with single avatars (like Solid Snake or Lara Croft) create the most personally felt attachment, but games that allow the fine-grained manipulation of multiple elements across a wide space (like *Full Spectrum Warrior*, *Civilization*, or *Rise of*

Nations) allow for a more “god’s eye” inspection of one’s manipulation, at the cost of personal attachment (perhaps), but with the gain of a wider “reach”. Such games “widen vision”, perhaps at the cost of intimacy. Neither sort of game is better or worse, they just have different effects and are useable for different purposes.

Property 3: Gaming as Experiential Learning with All the Right Conditions for Learning from Experience Met

Games put players in worlds where they experience things. This seems pretty simple, but it is, in fact, the foundation of how games recruit good learning. To see this, I need to talk briefly about contemporary research on human learning. Earlier learning theory argued that the mind works like a calculating device, something like a digital computer. On this view, humans think and learn by manipulating abstract symbols via logic-like rules.

Newer work, however, argues that people primarily think and learn through *experiences* they have had, not through abstract calculations and generalizations (Barsalou 1999a, b; Clark 1993, 1997; Gee 1992, 2004; Glenberg 1997; Hawkins 2005). People store these experiences in memory—and human long-term memory is now viewed as nearly limitless—and use them to run simulations in their minds to prepare for action and problem solving in new situations. These simulations help them form hypotheses about how to proceed in the new situation based on past experiences.

However, things are not quite that simple. There are conditions experiences need to meet to be truly useful for learning (diSessa 2000; Gee 2004; Kolodner 1993, 1997, 2006; see especially Kolodner 2006, p. 227, whose account I follow here). First, experiences are most useful for future problem solving if the experience is structured by

specific goals. Humans store their experiences best in terms of goals and how these goals did or did not work out. We have already argued this is core to gaming, since gaming is about discovering how **the rules of a game can be used to a player's advantage to accomplish the player's goals.**

Second, for experiences to be useful for future problem solving, they have to be interpreted. Interpreting experience means thinking—in action and after action—about how our goals relate to our reasoning in the situation. In gaming this is just the requirement that good games make that players think strategically—that they see through the “eye candy” to patterns and rules (what I will call “effectivity-affordance pairings” below) that will allow them to solve ever more challenging problems as they move through the game's levels.

Third, people learn best from their experiences when they get immediate feedback during those experiences so they can recognize and assess their errors and see where their expectations have failed. It is important, too, that they are encouraged to explain why their errors occurred and their expectations failed and what they could have done differently. Many games give good immediate feedback in terms of players being able to see and judge the results of their actions moment by moment and in terms of the sort of after-play assessments (in terms of graphs and charts) that real-time strategy games like *Rise of Nations* give players when they have finished a session of game play. Games, of course, do not require players to offer explanations for errors and expectation failures, but the social practices connected to multi-player gaming often do.

Fourth, learners need ample opportunities to apply their previous experiences—as interpreted—to new similar situations, so they can “debug” and improve their

interpretations of these experiences, gradually generalizing them beyond specific contexts. Of course, in good games, good level design pretty much ensures that this condition is met, as the player faces bosses and moves across the levels of the game, where later levels test previous skills, demand their mastery, and introduce new skills that must be integrated with old ones.

Fifth, learners need to learn from the interpreted experiences and explanations of other people, including both peers and more expert people. Social interaction, discussion, and sharing with peers, as well as mentoring from more advanced others, are important. Debriefing after an experience—that is, talking about why and how things worked in the accomplishment of goals—is important, as well. Again, games themselves and by themselves don't meet this condition, but gamer communities often do. It is interesting to note that the Army, in using games for training, often requires collaboration in play and debriefing afterwards.

Humans learn from experience. And these are the conditions experience must meet for effective and deep learning. Games, and the social practices and communities that accompany them, often meet these conditions pretty well.

Property 4: Gaming as Finding and Using Effectivity-Affordance

Matches between Bodies or Tools and Worlds.

Games create a “match” between affordances and effectivities. Let me explain what I mean by this: An “affordance” is a feature of the world (real or virtual) that will allow for a certain action to be taken, but only if it is matched by ability (called “effectivity”) in an actor who has the wherewithal to carry out such an action (Gibson, 1979). For example, in the massively multiplayer game *World of Warcraft* stags can be

killed and skinned (for making leather), but only by characters that have learned the Skinning skill. So a stag is an affordance for skinning for such a player, but not for one who has no such skill (no such effectivity). Some creatures in the game are not an affordance for skinning for any players, since they cannot be skinned at all. Affordances are relationships between the world and actors (or, as we will see below, between tools and actors).

Let's first consider games that give the player an avatar that serves the player in the virtual world as a surrogate body. Take, for example, a game like one in the *Thief: Deadly Shadows*. When you play this game, you get a surrogate body—namely Garrett, the master thief—that you as a player move around the virtual world and which you use to solve problems in that world. You also get a world—in this case a sort of medieval world of courtyards, towns, and castles. However, when you play as Garrett you realize that he has certain skills—he is good at some things (i.e., sneaking and hiding) and not so good at others (e.g., fighting out in the open). As a player, of course, since Garrett is your surrogate body in the virtual world, you inherit these skills.

Inheriting Garrett's skills means that if you want to solve problems in the game's world—that is, “win the game”—you have to look at the world in a specific way, not as “eye candy”, but as patches of light and dark and hidden nooks and crannies and edges and ledges that allow you to sneak and hide out of sight. Of course, the game's world is designed in such a way that the world can be readily seen and used in this way.

If you do see the virtual world of the game in this way—and use Garrett's body and skills well and appropriately—you get a perfect match between body (Garrett's, built for sneaking and hiding) and world (seen effectively as places for sneaking and hiding).

Of course, as player, you can, if you like, go against the grain of this match between body and world—you can, for instance, seek out well-lit open places and have Garrett directly attack foes. You can (sort of) succeed this way, in fact, but it is a continuous and frustrating struggle.

In games with a surrogate body that you micro-control—or in games like *Full Spectrum Warrior* where you direct a squad of surrogate bodies as a group without being any one of them—there is always a match to be found between the surrogate body (or bodies) and the world. In *Full Spectrum Warrior* you must see the world as “cover” and move your squad carefully from cover to cover so they are never in danger—and their bodies and skills are perfectly suited for such movements and maneuvers (thanks to artificial intelligence). *Thief* shows you what the world looks like to a master thief and *Full Spectrum Warrior* shows you what it looks like to a combat soldier, and *SWAT 4* shows you what it looks like to a SWAT team member.

Garrett, the soldiers in *Full Spectrum Warrior*, and the policemen in *SWAT 4* have skills that players do not initially have. The player gets to participate in—to watch and use and think with—their skills thanks to having control over their bodies. This is a new thing in the world: an incompetent beginner gets to control a competent body. Of course, I, the player, can send the body into harm’s way. I can misuse or squander its skills. However, as I learn to leverage the body-world match—to get Garrett sneaking and hiding in good synch with his world, to get the *Full Spectrum Warrior* soldiers moving from cover to cover in good synch with their world—I solve problems and learn to see the world (in this case a virtual world) in a new way. Anyone who has played a *Thief*: *Deadly Shadows* or *Full Spectrum Warrior* or *SWAT 4* (or *Chibi-Robo* for that matter)

knows that once you have become adept at the game, you can, in fact, even look at the real world in the same way as you have learned to look at the virtual world.

Students in school cannot use their teacher's competent body (or mind) to see the world in a certain way (e.g., to do a certain type of biology) and solve problems from that perspective. However, Garrett and the *Full Spectrum Warrior* soldiers and the *SWAT 4* policeman can be viewed in a different way. They can be seen, not just as surrogate bodies, but as tools—smart tools—tools that store knowledge and allow it to be leveraged and used. I, the player, use Garrett (with his built in skills for sneaking and hiding) to solve problems in the virtual world of the game. And I do this by finding and using the match between the knowledge and skills built into Garrett and the affordances built into the world of the game.

When we view Garrett as a tool we see something interesting. Seeing Garrett as a tool (a sneaking, hiding tool) means we have to see his world in terms of affordances for solving the game's problems in certain ways (i.e., through sneaking and hiding) and not others. This means we see the game world not in terms of its "eye candy," but now in a more abstract way. We see the graphically realistic and detailed world of the game as designed for sneaking and hiding. Its affordances for sneaking and hiding are foregrounded and other elements (pretty though they are) are backgrounded. Garrett's effectivities for hiding and sneaking are foregrounded and other aspects of Garrett—many of which are quite interesting—are backgrounded. The (virtual) world is now, in this sense, a more abstract place/space.

Games like *Civilization* and *Rise of Nations*, which let the player micro-control a great many game elements, move away from an avatar to a set of tools (e.g., the soldiers,

workers, fields, buildings, resources, vehicles, and so forth, of *Rise of Nations*). The player searches for a match between the effectivities of these tools (separately and as a set) and the affordances in the (virtual) world. The tools are “smart tools”; since they have knowledge and skills built into them that, in fact, constitute their effectivities (what they are “good” for, what affordances they can actually use). For example, workers know how to gather resources or build buildings. This can, of course, get to be quite complex and quite abstract. It is pretty clear in *Civilization* or *Rise of Nations* that you are carrying out plans, engaging in tactics and strategies, building economies and futures, not just moving avatars. However, this movement from avatar to tools is really a step on a continuum, since, as we have just seen, Garrett or Lara Croft can be viewed both as a surrogate body and as a smart tool.

Property 5: Gaming as Modeling and using Models to Make Learning from Concrete Experience More General and Abstract

I have just pointed out that games require players to look through “eye candy” to find effectivity-affordance matches. Players have to look at the game world in a certain way that fits with the body, bodies, or tools they have been given to micro-manipulate. This renders a concrete experience—the experience of acting in and on a concrete real-looking world—somewhat more abstract, as one looks at the world as a “system”.

This combination of concrete experience and a more abstract view is crucial to learning. I have said above that humans learn (and think) through their embodied experiences, provided certain conditions are met. However, learning through experiences—experiential learning—has a problem: it can be too concrete, too tied to specific situations, not general enough.

I have already argued that this problem, as far as games are concerned, is partly solved by two things: first, the need to look for—and see the game world in terms of—effectivity-affordance pairings, and, second, the ways in which the conditions for learning from experience—conditions that require reflection, interpretation, and strategy, as well as comparing and contrasting multiple experiences (conditions which we discussed in Section 3 above)—are met in good games. However, the problem of too much concreteness is solved in games in another way, as well, one that is quite powerful.

Even though players are experiencing a game’s virtual world—which might be quite graphically detailed; very often gamers are using and thinking in terms of “models”. And models are crucial for good learning (diSessa 2000, 2004; Lehrer & Schauble 2000, 2005, 2006; Nersessian 2002). They help bridge between concrete experiences and more abstract and systematic understandings. Models are crucial to games and gaming, as well.

Models are just depictions of a real thing (like planes, cars, or buildings) or a system (like atomic structure, weather patterns, traffic flow, eco-systems, social systems, and so forth) that are simpler than the real thing, stressing some properties of the thing and not others. They are used for imaginative thought, learning, and action when the real thing is too large, too complex, too expensive, or too dangerous to deal with directly.

Consider a model plane. A model plane closely resembles the thing it is modeling (a real plane). It could be used by a child for play or by a scientist studying aerodynamics. But models don’t have to closely resemble what they are modeling. In fact, models can be arranged on a continuum of how closely they resemble the thing they are modeling. They can be more or less “abstract” or “concrete.” One model plane may

have lots of details. Another may be a simple balsa-wood wings and frame construction, no frills. Even more abstractly, the blue print of the plane, on a piece of paper, is still a model, useful for some purposes (e.g., planning and building) and not others. It is a model that resembles the plane very little, but still corresponds to the real plane in a patterned way. It's an abstract picture.

We can go even further and consider a model of the plane that is chart with all the plane's different parts listed down a set of rows and a set of numbers ranged along the top in columns. The intersection of a part and number would stand for the amount of stress each part is under in flight. For each part we can trace along the row and see a number representing how much stress this part is under in flight. No resemblance, really, left here, but the chart still corresponds to the plane. We can still map from pieces of the chart to pieces of the plane. The chart still represents some properties of the plane, though this is a very abstract picture of the plane, indeed, and one useful for a narrow purpose.

However, this type of model—at the very abstract end of the continuum of resemblance—shows us another important feature of models and modeling. Such a model captures an invisible, relatively “deep” (that is, not so readily apparent) property of the plane, namely how parts interact with stress. Of course, we could imagine a much more user-friendly picture (model) of this property, perhaps a model plane all of whose parts are color coded (say in degrees of red) for how much stress they must bear in flight. This is more user-friendly and it makes clear the mixture of what is readily apparent (the plane and its parts) and what is a deep (less apparent) property, namely stress on parts.

These are very basic matters. Models and modeling are basic to human play. They are basic to a great many other human enterprises, as well, for example, science (a diagram of a cell), architecture (model buildings), engineering (model bridges), art (the clay figure the sculptor makes before making the real statue), video and film (e.g., story boards), writing (e.g., outlines), cooking (recipes), travel (maps), and many more.

Models are basic to video games, as well. Some video games are simulations in which the player is inside the simulation thanks to the presence of an avatar. And, of course, all simulations are models of what they are simulating. So *World of WarCraft* simulates (models) a world of mountains, lakes, roads, buildings, creatures, and so forth, that, while fantasy, is meant to resemble aspects of the real world. However, players for the most part pay very little attention to this modeling aspect of *World of WarCraft*, because it usually plays no important role in game play. Rather, players concentrate on the embodied experiences of play, problem solving, and socialization that *World of WarCraft* offers. By and large, the fact that it models environments does not matter all that much to the game play.

However, sometimes in *World of WarCraft* this is not true; sometimes the modeling aspect comes to the fore. For example, when I get stuck trying to walk up the inclines and crevices of a mountain in *World of WarCraft*, I begin to think about how the game's mountain is representing (modeling) gravity and resistance in the real world, sometimes with anger, because I realize that it did not model them well enough to ensure that I can get up an incline that in the real world I could, but in the game I can't. In other games, where one's character seems more than tall enough to jump over an obstacle, but can't, the player is well aware the model is a model and isn't working well. So, in games

like *World of WarCraft* the modeling aspect comes to the fore only when there are problems.

However, there are other games in which the modeling aspect of the simulation is crucial. Players in these games are having experiences, just as they are in *World of WarCraft* or *Half-Life*, but the modeling aspect is also crucial at nearly all points, not just intermittently. In a game like *Civilization*, for instance, the depictions of landscapes, cities, and armies are not very realistic, not nearly as realistic as in *World of WarCraft*. For example, in *Civilization*, a small set of soldiers stands for a whole army and the landscape looks like a colorful map. However, given the nature of game play in *Civilization*, these are clearly meant to be models of real things, stressing only some of their properties. They are clearly meant to be used for quite specific purposes in the game, for example, modeling large scale military interactions across time and space and modeling the role of geographical features in the historical development of different civilizations.

However, even in games where, at the big picture level, modeling is not integral to game play in terms of their overall virtual worlds—games like *World of WarCraft* or *Half-Life*—very often models appear ubiquitously *inside* the game to aid the player's problem solving. For example, most games have maps that model the terrain (and maps are pretty abstract models) and allow players to navigate and plan. The bottom of *World of WarCraft*'s screen is an abstract model of the player's abilities and skills. Lots of games allow players to turn on and off a myriad of screens that display charts, lists, and graphs depicting various aspects of game play, equipment, abilities, skills, accomplishments, and other things. In a first-person shooter, the screen that shows all the

guns a player has, their firing types, and their ammunition is a model of the game's weapon system, an abstract picture of it made for planning, strategizing, predicting, and problem-solving.

Models inside games go further, much further. Players and player communities often build modifications of games that are models used to solve certain sorts of problems. For example, *World of Warcraft* players can download a mod that displays a chart (during actual fighting) that lists each player's class (e.g., Druid, Priest, Warrior, Mage, Paladin, etc.) and the amount of damage they are doing in a group raid inside a dungeon. This chart ("damage meter") can be used to check—publicly—that each player is holding up his end of the group task (so Warriors better be doing lots of damage and healing ("holy") Priests better not be—they had better be concentrating on healing rather than attacking). This is one of several models, almost all of them made by players, that help players solve a very real-world problem, namely the problem of individuals attempting to take a free ride in a group or attempting to hide their lack of skill. At the same time, such models generate a good deal of debate on fan forums about how good they are and how they should or should not be used.

Models and modeling reach a new pitch in games like those in the *Tony Hawk* series. First, each game is a model of the practices and culture of skateboarders. Within that larger model, there are a myriad of models of boards, dress styles, tricks, and parks. However, players can readily design their own skaters, clothes, boards, tricks, points for tricks, and skate parks. That is, they can build their own models. When they build a model skate park, they interact with a set of more abstract models of environments (screens made up of grids and rotatable objects) that help them build the more specific

and realistic looking model skate park they want (like a toy plane). Indeed, as skating in the real world changes, the models in the game and those made by players change, each time trying to capture things that are seen as important or essential, all the while balancing a variety of criteria about fidelity to different things and systems. This is modeling with a vengeance. Here modeling is integral to game play at all levels and in every way.

Models and modeling are important to learning because, while people learn from their interpreted experiences, as we have argued above, models and modeling allow specific aspects of experience to be interrogated and used for problem solving in ways that lead from concreteness to abstraction (diSessa 2004; Lehrer & Schauble 2006). Models and modeling are important to game design because in-game models are tools to facilitate, enrich, and deepen the problem solving the game designer is building.

Property 6: Games as Player-Enacted Stories or Trajectories

There has been much controversy over the role of story in games. And, of course, many (but by no means all) video games have stories much in the way in which books and movies have stories—for example, the *Final Fantasy* games or the *Metal Gear Solid* games. This is what I will call the “designers’ story:” The player has not made this story up, the games’ designers have. I do not want to enter here into the controversies over the role of such stories in games, save to say that there is a second story in games that is, in my view, more important to game play than is the designers’ story.

To see this second story, consider a game like *Castlevania: Symphony of the Night*. Any one who has played this (great) game and who does everything you can do in the game will, in the end, have done all the same things as any other player. A player

who does less will have done some sub-set of this. This is just like a book. Everyone who reads the whole book will have read the same text. Any reader who reads less will have read some sub-set of this whole text.

However, each player of *Castlevania* will have done and found things in entirely different orders and in different ways from each other. Players will have ventured into the parts of the castle in different orders, they will have revisited them a different number of times. They will have faced the bosses in the game at different times and will have defeated them in different ways and with different degrees of difficulty. They will have found key items in the game in different orders. They will have made different choices of what strategies to use, when to save, and what equipment to wear and use. This is to say that each player has enacted a different *trajectory* through the game.

There is no sense (or not much of one, or not one in the same sense) of different trajectories in a game like *Tetris*. What allows each of us to feel and recognize a different trajectory in a game like *Castlevania* is that such games are composed of events that we, as players, created and set into motion. We can recognize that a distinctive event (e.g., Me as Alucard killed his/my first Sword Lord) happened before or after another distinctive event (e.g., Me as Alucard who found the gold ring). Such events give the player a way to mark time and against this marking each player comes to see that he or she has enacted a unique trajectory through the game space.

This trajectory has an important consequence. Your Alucard is different than my Alucard. Yours has had a different trajectory than mine. This means that the virtual character in the game world—Alucard in this case—is different for each player in a significant and meaningful way. The hero is, thus, not Alucard from the designer's story,

nor is it you the real-world player (you are, after all, playing Alucard). It is “Alucard-you,” a melding of the virtual character, Alucard, and you, the real-world player who has steered Alucard on a unique trajectory through the game.

The hero in my own personal trajectory through the game was “Alucard-Jim,” a blend between a virtual (Alucard) and real person (me). This is why players can so readily switch between saying “Alucard killed the Sword Lord” and “I killed the Sword Lord.” The real actor here is a composite or blend: Alucard-you (me).

This trajectory is the second story. Let’s call it, to distinguish it from the designer’s story, the “trajectory story”. This is the important story in *Castlevania*. Players can play the game over again to gain another trajectory—good games lend themselves to such re-play, to the building of new trajectories. This trajectory is personal and individual in a game like *Castlevania*. It can be both personal and social in a multiplayer game.

So when I play *Castlevania*, I generate a unique story—the trajectory story. This story is the enacted tale of Alucard-Jim and I can lard it up with all the fantasies, values, and morals I want to—no permissions needed, no critics allowed.

In the Alucard-Jim story, Alucard-Jim was only able to beat the large knight with the owl on his shoulder at the front of the castle (the Owl Knight) after Alucard-Jim became more powerful (Alucard by gaining experience and Jim by getting more practice and skill). For some other player, let’s say “Jane,” Alucard-Jane had a much easier time killing the Owl Knight early on with less hassle and effort than did Alucard-Jim. Alucard-Jim had, in fact, tried unsuccessfully several times earlier to kill the Owl Knight.

After Alucard and Jim had gained enough experience, Alucard-Jim proudly marched to the front of the castle and, with great glee, mastered him easily.

This event (Alucard-Jim finally kills the Owl Knight at the front of the castle) became one of my own unique high points in the story I was performing in playing the game. Though the designers' own the designer's story, I own the Alucard-Jim story, the trajectory story, which has its own unique high (and low) points. Jane has a different trajectory story. Each of us human beings has a unique trajectory through life. Indeed, the trajectory (second story) I am talking about in *Castlevania* is much more similar to our own life trajectories than it is to the linear and intricately pre-designed stories in books and movies.

The trajectory story is apparent and immediate in games with an individual avatar (like Alucard), whether this is played in the first or third person. But the effect—with different nuances—is present in other sorts of games, as well. In *Full Spectrum Warrior*, the player still says things like “we lost”, identifying with the team (the squads he or she controls). In *Civilization*, players identify with their civilization. In both cases—*Full Spectrum Warrior* or *Civilization*—players have unique trajectories through the games. So, too, for yet other types of games, as long as there are player enacted events that can be lined up in time.

The trajectory a player takes through a game—the virtual-real story—can, in certain circumstances, give space a special sort of deep meaning in a game. If I can revisit spaces (places) in a game, and different things happen there at different times (e.g., I have different experiences with the Owl Lord each time I go to the front of the castle in *Castlevania* or I have different quest and social experiences in the Bone Wastes in *World*

of *WarCraft* over time), then there are layers of meaning (layers of my trajectory story) laid down, one on top of the other, at that place.

Space becomes a patchwork of such meaning-layered places, connected in a myriad of ways through the meaningful (storied—in the trajectory sense) connections across layers (this event that happened *here* is connected, in some fashion, with that event that happened *there*). Anyone who has played *World of Warcraft* a great deal has had this feeling of layered and connected space as they fly over regions of the game, looking down at a now fully storied space—storied in the sense of my trajectory story. Connections within and across layers can be meaningful in many different ways to players, most certainly including emotionally meaningful.

Conclusion

I restate the deep learning properties of good video games here in terms of questions. Deep games—commercial or “serious”—not only have these properties, but implement them powerfully. There are, of course, perfectly entertaining games that meet none of these conditions—for example, the delightful *Sam and Max* games do not. Games do not have to be long and complicated to meet the conditions—*Diner Dash* meets them and is neither. *Sam and Max* probably helps people with mental alertness and general problem solving skills, but *Diner Dash* does more—in addition, it makes the player embody and empathize with a set of connected problems (a problem space) connected to a certain “identity” or “way of being in the world”. So does *Thief: Deadly Shadows*, *Full Spectrum Warrior*, *SWAT4*, *Civilization 4*, and *Rome: Total War*. A game like *The Sims* invites players to create these properties for themselves, offering them resources that allow them to do so in powerful and entertaining ways.

For me, it is an interesting question to ask if we can make “serious games” beyond games like *America’s Army*, *Full Spectrum Warrior*, and *SWAT 4* (and their more official training versions), games that focus on armed conflict or controlling armed conflict. At their best the properties below allow players to have powerful experiences that compete with experience in the real world precisely because experiences in the real world, at their best—when we humans feel control, agency, deep learning and mastery—meet just these properties. But that is a story for another day.

- Property 1: Does game play allow and encourage the player to “psych out” and take advantage of an underlying rule system to accomplish personally held goals to which the player is emotionally attached?**
- Property 2: Does the game allow the player micro-control that creates either a sense of embodied intimacy or a feeling of reach in power and vision?**
- Property 3: Does the game offer the player experiences that meet the conditions for good learning (discussed above)?**
- Property 4: Does the game allow, encourage, and help players find and use effectivity-affordance matches between smart bodies or tools and worlds?**
- Property 5: Does the game use modeling or models to make learning from experience more general and abstract?**
- Property 6: Does the game allow and encourage the player to enact his or own unique trajectory through the game, thereby creating his or her own story?**

REFERENCES

Barsalou, L. W. (1999a). Language comprehension: Archival memory or preparation for situated action. Discourse Processes 28: 61-80.

Barsalou, L. W. (1999b). Perceptual symbol systems. Behavioral and Brain Sciences 22: 577-660.

Bransford, J., Brown, A. L., and Cocking, R. R., (2000). How people learn: Brain, mind, experience, and school: Expanded Edition. Washington, DC: National Academy Press.

Clark, A. (1993). Associative engines: Connectionism, concepts, and representational change. Cambridge: Cambridge University Press.

Clark, A. (1997). Being there: Putting brain, body, and world together again. Cambridge, Mass.: MIT Press.

Damasio, A. (1994). Descartes' error: Emotion, reason, and the human brain. New York: Penguin.

Damasio, A. (1999). The feeling of what happens: Body and emotion in the making of consciousness. Orlando, FL: Harvest Books.

- Damasio, A. (2003). Looking for Spinoza: Joy, sorrow, and the feeling brain. Orlando, FL: Harvest Books.
- diSessa, A. A. (2000). Changing minds: Computers, learning, and literacy. Cambridge, Mass.: MIT Press.
- diSessa, A. A. (2004). Metarepresentation: Native competence and targets for instruction. Cognition and Instruction 22: 293-331.
- Gee, J. P. (1992). The social mind: Language, ideology, and social practice. New York: Bergin & Garvey.
- Gee, J. P. (2003). What video games have to teach us about learning and literacy. New York: Palgrave/Macmillan.
- Gee, J. P. (2004). Situated language and learning: A Critique of traditional schooling. London: Routledge.
- Gee, J. P. (2005). Why video games are good for your soul: Pleasure and learning. Melbourne: Common Ground.
- Gee, J. P. (2007). Good video games and good learning: Collected essays on video games, learning, and literacy. New York: Peter Lang.

Gibson, J.J. (1979). The ecological approach to visual perception. Boston: Houghton Mifflin.

Glenberg, A. M. (1997). What is memory for. Behavioral and Brain Sciences 20: 1-55.

Hawkins, J. (2005). On intelligence. New York: Henry Holt.

Kolodner, J. L. (1993). Case based reasoning. San Mateo, CA: Morgan Kaufmann Publishers.

Kolodner, J. L. (1997). Educational implications of analogy: A view from case-based reasoning. American Psychologist 52: 57-66.

Kolodner, J. L. (2006). Case-based reasoning. In R. K. Sawyer, Ed., The Cambridge handbook of the learning sciences. Cambridge: Cambridge University Press, pp. 225-242.

Lehrer, R. & Schauble (2000). Modeling in mathematics and science. In R. Glaser, Ed., Advances in instructional psychology: Educational design and cognitive science. Vol. 5. Mahwah, NJ: Lawrence Erlbaum, pp. 101-159).

Lehrer, R. & Schauble, L. (2005). Developing modeling and argument in the elementary grades. In T. Romberg, T. P. Carpenter, & F. Dremock, Eds., Understanding mathematics and science matters. Mahwah, NJ: Lawrence Erlbaum, pp. 29-53.

Lehrer, R. & Schauble, L. (2006). Cultivating model-based reasoning in science education. In R. K. Sawyer, Ed., The Cambridge handbook of the learning sciences. Cambridge: Cambridge University Press, pp. 371-387.

Nersessian, N. J. (2002). The cognitive basis of model-based reasoning in science. In P. Carruthers, S. Stich, & M. Siegal, Eds., The cognitive basis of science. Cambridge: Cambridge University Press, pp. 133-155.

Sawyer, R. K., Ed. (2006). The Cambridge handbook of the learning sciences. Cambridge: Cambridge University Press.