PsychNology Journal, 2002
Volume 1, Number 1, 57 - 70

Virtual Unreality of Videogames

Guardini Pietro♠

Department of General Psychology

University of Padova, Italy

ABSTRACT

Beginning with an objection raised to the excessive attention promotion gives to the aspect of "realism" in videogames, I chose to describe instead how excessive realism may be counterproductive. I identify and focus on what game designers wish to obtain from their products, then proceed to introduce and explain the concept of "virtual unreality of videogames". A number of examples and counter-examples serve to describe important principles used by game designers to build what I have called virtual unreality.

Keywords: *videogames, usability, realism, game design, virtual environments*

Received 2 July 2002; received in revised form 14 October 2002; accepted 15 October 2002.

1. Long live realism!

According to the quotes printed on any videogame package found on the shelves of any store, getting as close as possible to reality seems to be their main objective: "If it is possible in the real world, you can also do it here", "Realistic car handling and real damage!", "Artificial Intelligence extremely real!", "Real Race Tracks and Cars", and so on. Be it racing games or sport simulations, the whole videogame industry's objective appears to be the construction of a product as close to reality as possible; after all, "realism" is one of the parameters most widely used by game-magazine journalists to rate videogames, and it is also one of the most commonly-found words in several websites where new or still-in-development titles are advertised. For instance one of the most successful racing games for Sony Playstation is called "*Gran Turismo: the real driving simulator"* (Sony / Polyphony Digital Entertainment,

♠ Correspondig Author:
Pietro Guardini, Dipartimento di Psicologia Generale,
Università degli Studi di Padova, via Venezia 8, 35131 Padova, Italy.
Tel: +39-049-8276901
Fax: +39-049-8276600
E-mail: pguardini@yahoo.it

1998). It is quite odd that the race for realism is joined even by products inspired in science fiction: for example, space-combat videogames or ones that include characters without any direct reference in the real world, like dinosaurs.

2. How to get closer to reality?

If the word realism applied to videogames is defined as the ability of the videogame to present places, situations and themes that have a clear referent in the real world, and therefore are seen as something that the player / buyer will already be acquainted with, the following question may be asked: how do game designers try to get closer to reality?

3. Graphics & Sound

The first way in which verisimilitude may be obtained is through the graphic aspects: accuracy in the reproduction of objects (such as buildings, vehicles, and characters) is possible on two levels. The first concerns the designing of a model of the represented object ("modelling"), which is constructed by using differently-shaped polygons arranged like building bricks, to occupy a volume in the three-dimensional virtual space. The second concerns textures ("texturing"), that is, images used like wallpaper would be, to coat the outer (and all visible) faces of the model; the photorealistic aspect of three-dimensional objects found in many contemporary products thus depends largely on this element.

In the world of videogames, the modelling process in some cases has required the use of modern and sophisticated technologies, such as Global Position System (GPS) – for instance, to build racing circuits in *F1 Racing Championship* (Ubisoft 2001) – or laser scanning – for car-model construction in *Toca Touring Car Championship* (Codemasters 1997). Through these technologies, a scaled digital clone of the object with extreme precision may be obtained. The texturing process has long been based on the manipulation of actual high-resolution digital photography of the objects to be built. The designer team of the marvellous *Max Payne* (God Games / Remedy 2001), which masterfully tells a story set in New York, revealed they constructed their textures from many pictures taken around the city's different neighbourhoods (Määttä, 2002).

Software houses and publishers spend millions of dollars to secure the official license from FIA (Federation Internationale de l'Automobile) for a Formula 1 videogame, since the license grants them access to the use of the names and faces of actual pilots, as well the appearance of actual championship cars, the logos of sponsors and official racing circuit.

A similar thing applies to the obviously more exclusive world of military and civil flight simulators. In the first case, even the US Air Force was recruited as a collaborator to one product (*U.S. Air Force*, Electronic Arts / Jane's Combat Simulations, 1999), while the second market has virtually been cornered by Microsoft's Flight Simulator (2002, latest version), which sports the logos and sponsorship of impressive firms such as Cessna, Boeing, Bell, and Learjet.

The sound receives no less attention. In car-racing simulators, the noise of engines is usually recorded from real cars, and then reproduced to allow for a wider range and a more accurate characterization of the different types and powers of engines. For example, the presence of the turbo feature in the engine of any car model in the game "*Need for Speed – Porsche 2000"* (Electronic Arts, 2000) may be detected by simply revving the engine – not unlike real life.

4. Dynamics & AI

Once the objects have been built, rules of behaviour are required for the objects if a realistic behaviour is wished. For example, actual physical laws and limitations are resorted to in order to obtain credible movement of any "in game" object, and AI routines have to be inserted to give both motion and motivation to those objects and characters that are outside the player's direct control. For instance, in a football videogame it is to be expected that the ball move as we would see it in a stadium, and that the opposing team does not merely converge on the ball, but organizes to try to steal it away from the human player and score a goal of their own.

5. The point: even better than the real thing

It seems therefore that anyone who buys a videogame will bring home a piece of reality, thus being able to live out experiences that for various reasons may be beyond their reach - a student at the end of the semester will be highly unlikely to be driving a Lamborghini Murciélago along the French Riviera, just as a lawyer with a passion for trains will never be able to pilot the Orient Express, or any more than a retired gardener will be able to enter the skin of Julius Caesar during his Gallic war campaigns…

But this is not the case at all: game developers are well aware of the fact that the final objective towards which they employ their skills has nothing to do with the provision of a duplicate of reality. If this were the case, winning a rally videogame would require the same expertise needed to win a real-life rally. This would mean years of training, dedication and familiarity with the medium, while the videogame player wishes to learn to drive in a few minutes, and to enjoy himself by winning and driving in

spectacular fashion while spending just a few minutes in picking up the relevant skills. A 1:1 relationship between game and reality is not a viable option: a player may win a rally in a videogame even though he may not even have a driving license, because the two things – real-life driving and game-driving – are completely different, and their objectives are different too. The main objective of a videogame is to satisfy a need for entertainment.

The task of any game designer is therefore the construction, within his videogames, of that which I will call "virtual unreality", that is the illusion of a world, a quasi-reality. Its reality is only superficial and apparent, since the videogame world is an artefact, created with the specific goal of fulfilling and even anticipating the expectations of the player at any given moment. Virtual unreality is therefore a less-immediate aspect of the more general concept of virtual reality, bearing in mind that virtual digital worlds are the issue in question.

6. It's not a hardware question

The reasons underlying the difference in objectives adopted in the contextualized settings in videogames (driving a car, playing football, piloting a plane, etc.) are not, as one may think, linked to hardware limitations. The problem does not lie with technical difficulties, such as calculus capacity, the use of specialized hardware to reproduce effects on peripheral control devices (vibrations of the steering wheel due to road conditions, air current that make it more difficult to hold the cloche steady, etc.) or to the player himself (seat vibration, wind in the hair, changes in proprioception).

After all, the availability of peripherals that guarantee a multisensorial feedback has increased constantly: steering wheels, handlebar controllers and joypads with forcefeedback have become the norm after a few years of uncertainty due to high prices and rudimentary technologies. More recent and less widespread products are also available, such as a seat with embedded hydraulic jacks that rotates and inclines, involving the player in its movements, the control peripherals and the computer monitor. The experience is not unlike a ride on a mechanical bull, enhanced by the constant menace that the monitor may become unscrewed from its support thus resulting in multiple facial lacerations – but then again maybe this is just a personal phobia… The "I-smell" technology, which adds to the computer a peripheral that emits different fragrances depending on what the monitor is showing was also the source of much interest, until the lack of funding interrupted any further R&D on the project.

7. Examples

A number of examples supporting the statements that game programmers do not look for a slavish adherence to reality:

Simon Kershaw was the producer of *F-22 Air Dominance Fighter* (Ocean / DID, 1997), by Digital Image Design, a British software house that produced flight simulators both for the military and the home entertainment markets. In an interview he gave in 1997 to the top-selling Italian videogame magazine *The Games Machine* (Silvestri 1997), in answer to the question "Wich are the differences between your flight simulators and videogames?", he admitted that the differences are remarkable, in particular as concerns the behaviour of the plane and the physics of a simulated world. The simulated plane, a modern F-22 fighter, has very efficient computerized systems that would allow the plane to keep its stability even in the event of serious damage to one wing caused by an enemy missile. According to Kershaw, replicating such a device in a videogame would surprise its buyer, since the public paradoxically would not accept it as realistic. Moreover, fights usually occur between planes who are hundreds of miles apart: the enemy is a small dot on a radar, and the range of missiles makes it unnecessary to be close (dogfights, that is, close-up duel between planes, are considered as too dangerous, and the damage caused by short-range weapons, such as a machine gun, is slight if compared to missiles of the "fire and forget" kind). For videogame players, a game structured around the exchange of missiles until the small dot on the radar disappears or the player is eliminated must be highly unappealing, and even more so if he has just watched the movie *Top Gun*, where the planes brush against each other while exchanging machine-gun fire. This is why the programmers have altered the range of missiles, in order to force the planes to draw nearer to their objective before releasing the missiles, a gimmick that makes the fights more challenging.

Also, the stealth devices that make planes invisible to radars in reality would make any game too simple and boring, since the player would be able to easily reach his objective without any counteroffensive, and without any confrontation with enemy planes. According to the development team, this problem has been solved in two different ways. Interactive Magic game designers (*iF-22*, 1997) intervened directly on the plane's capabilities, so that its stealth functions would be less effective; DID programmers left unchanged the plane's parameters, but made it necessary for the plane to exit the stealth mode in order to complete the mission (Silvestri, 1997).

A further example: In his excellent book *Trigger Happy - The Inner Life of Videogames* (2000), Steven Poole mentions the same kind of problem when talking of space-combat simulators, with a particular reference to laser fights. The most common representation of laser cannon consists of two brightly-coloured lines that, once the

trigger has been pulled, converge in one point, where the sight is. The key to victory is shooting through the viewfinder, managing to anticipate the enemy ship.

Already, two mistakes have been made here. Firstly, laser rays in space are not visible, since there is nothing on which the light beam may refract; the depiction of laser rays is therefore incorrect. Secondly, since laser is made of particles that travel at the speed of light, the reach their target almost instantaneously, unlike normal bullets in the Earth's atmosphere, therefore making it unnecessary to anticipate the movements of the enemy. The argument is undeniably persuasive; nonetheless, the decision to prove the thesis of the untrue realism of a videogame that depicts the player as sitting in the pilot's place of a stellar fighter does seem a bit odd.

One last example, also taken from Poole's book: Richard Darling, Creative Director of Codemasters, a British development and production house, in an interview with the author, reveals a device adopted by the game designers of his company to make racing games more fun and challenging: "If you are driving along and you crash, and the pack goes ahead of you, you won't necessarily notice if they all slow down a bit so you get a chance to catch them up, and you feel like you're still in the game – whereas a good player wouldn't have crashed in the first place, and so the cars wouldn't have slowed down, so you can have a competitive time either way" (Poole 2000: p. 186).

8. The exception that proves the rule

Products however do exist that display a high grade of realism and require a remarkable learning curve. The best example is *Grand Prix Legends* (Sierra / Papyrus 1998), which features the Formula 1 World Championship of 1967, including cars that, due to their strict adherence to reality, are particularly hard to handle, due to the power of engines and the lack of adherence to the tarmac of a road. This product required various days of practice in order to complete a few circuit round without accidents, compared to the standard ten minutes or so, and completing a race with a respectable classification may take weeks or even months. *Grand Prix Legends* never became a bestseller, but it does have a hardcore fan basis that has formed online gaming and information-exchange communities, where players may exchange information on the best setup for a certain car and are always ready to challenge each other online. Fans have always developed many add-ons and patches, that is, programmes that may be added to the original videogame, that enhance the use-value of the product introducing new cars, circuits, and details of vehicles and circuits.

9. Building worlds for Virtual Unreality: consistency, supporting design, environmental storytelling.

It is also peculiar that Chris Hecker should begin his article "Physics in computer games" (2000) by citing the very game that we have just been discussing, *Grand Prix Legends*. At the beginning of the he describes the gyroscopic effects of wheels and engine, that are at any moment being calculated in real time in order to influence the oscillations of the car's bodyframe – a testimony to the extent of the programmers' attention to the reproduction of physical behaviour of objects in this game. According to Hecker, videogames "physics simulators" – that is, "the part of the program that controls the physics of the environment, the very rules governing how the entities in the game interact" (p. 35) – have to become increasingly sophisticated to keep up-to-date; therefore, "game developers are turning to serious algorithms from the computer science research community" (p. 35). Being sophisticated however does not completely solve the problem: "the most important question for a game developer to ask is whether the physics simulator in a particular game is a good match for the design" (p. 36) of the game he's working on. Since technical restrictions mean that a family of algorithms that accurately simulate the real world does not exist yet, the game designer is forced to simplify the operation and select an algorithm able to foresee and control only those physical rules that he deems more relevant or indispensable to the functioning of the game.

This is a further example of how a virtual unreality is built, that is, of how game designers do not have the simple task of copying reality, but are forced to build one from scratch, while keeping their objectives present and having to make important choices. And it is precisely these choices who will determine the success of the product.

Hecker himself states that, contrary to what videogame advertising states, "the goal of the games is not reality itself, but the real goal is consistency" (p. 36). A consistent environment requires that the player be immersed into the virtual environment, a concept that in the academic environment is measured as presence. The simplification of physical laws inevitably leads to the appearance of imperfections and inconsistencies in the environment. For example, Richard Rouse III offered an instance set in the roleplaying universe, in which the player must solve a puzzle that requires him to open a door by activating a switch (Rouse, 2001). This is done applying pressure on a plate on the floor under which the device is hidden. To do so, he needs to set a number of objects on the plate, and it so happen that in the next

room he can find a mound of stones. The obvious solution is then to go back and forward between the rooms, carrying and dropping stones until the mecanism is actioned. The game designer has anticipated and inserted this solution in the game. However, the player may also opt for a different solution: dropping his various weapons over the plate, or, through a spell, summon monsters that will position themselves over the plate following the player's order. These are all logical solutions; however, they would only work if the programmer had inserted in the code that the plate may react by opening to objects other that rocks, such as the abovementioned weapons and monsters. If the programmer however has decided that only the weight of stones will open the door, then the player will face one of the inconsistencies of the virtual world ("How is it possible that my whole arsenal is not enough to make the device work?") and will be force to act in accordance with the will of programmers. This is the anticipatory school of game design, whereby the designer thinks what the player might do and develops (hardwires) the game to function only in accordance with those actions; this school is somehow surpassed by the vision of the complex systems school, of a more holistic nature. For example, if every object in the virtual world is endowed with characteristics that include the attribute "weight", then the player will be able to overcome the pressure-plate puzzle with whatever object he chooses, and will not be limited to the objects selected by the programmer. The development of such a complex system certainly requires greater efforts on the part of game designers, but it is certainly more gratifying both for them and for players to be able to find individual solutions puzzles, thanks to the freedom that the product offers.

This observation leads to important conclusions: the game programmer who resorts to a complex system enables the player to develop a gaming style of his own, and at the same time to avoid the pitfalls of environmental inconsistencies, strengthening the sense of immersion.

It is interesting to note that some of the gaming techniques that may be defined as emergent became very fashionable among players. This was exactly the case with *Quake II* (Activision / idSoftware 1997) and the technique of *rocketjumping*. The aim of this extremely popular first person shooter (FPS) is to get rid of a multitude of enemies by recurring to weapons of different power. Rocketjumping is achieved by picking one of the most powerful weapons (such as the missile launcher), and shooting towards the floor. The result is that the player is propelled upwards, but sustaining in the process a fair amount of self-inflicted damage. The strategy may seem counterproductive at first sight; however, it has been revealed as priceless in deathmatches, that is challenges between two human adversaries: taking a very high leap may enable the player to foil his opponent's attack, or reach strategic positions that would otherwise be

unattainable, from which the player's own attacks may be more successfully carried out. This videogame has evolved and has found a new dimension through the trial-and-error processes carried out by the players themselves, who thus contribute to and welcome the creation of a "virtual unreality".

In the aforementioned example of Quake, the game programmers offer players an avatar with human traits and characteristics, plus power-ups, that is, enhancements of strength, speed, resistance to blows, etc. What would happen if the character that the player controls had utterly different characteristics? This was the case for example with *Alien vs. Predator* and its sequel (Fox Interactive / Rebellion 1999; Sierra / Monolith 2002), games that allow the player to don the gear of an Alien, the selfsame creature that caused so much trouble to Ellen Ripley in the well-known series of films, or of a Predator. The Alien can scale walls like spiders, make huge leaps, have wide-angle vision, and are extremely agile. All these characteristics allows the game player to further engage in what is a clearly fantastic virtual unreality, obviously unlike the real world, but also force him to revise his gaming strategies.

10. Supporting Design

The aseptic and faithful reproduction of the environments that are found in the real world does not therefore seem to be as useful to videogame programming as may seem at first sight. We now may analyze a number of examples of how game designers, making the most of the enormous freedom at their disposal, build virtual environment that spoil the player and comply with his every desire in a hidden way, supporting and guiding him with an invisible hand that masks behind the control type, the environmental design, and the actions of non-playable characters, the evolution of the story arch and the puzzles to be solved. Most of the time, much of the success of a game lies in the ability of the game designer to produce something that avoids being too linear and proceeding along pre-established lines, but also does not give the player full freedom, a characteristic that may result in his feeling disoriented. The player should be able to enjoy "a sense of freedom accompanied by a sense of guidance" (Rouse, 2001). And it is precisely this sense of guidance - constituting part of the essence of the virtual unreality that is the subject of the present article – that is structured through a number of small gimmicks that most of the players will never be aware of, such as the ones that were given as examples. It is almost as though the player signed a contract with the game designers, whereby he will ignore inconsistencies and limits that these will impose on the game. In exchange, the game designer undertakes to create for the player the illusion of an experience that will be highly gratifying.

It is also remarkable to notice how a number of products owe part of their success precisely to the ironic exploitation of this "contract", bringing to the fore to nature of the videogame or purposefully breaking the player's sense of immersion. As example of the further, the intriguing *The Nomad Soul* (Eidos / Quantic Dream 1999), the computer is taken to be a portal to a different real world, where the soul of the player is taken and implanted in the body of a local; the player must then take control of the alien being, and win his own soul back through a number of missions and defeat the enemy. *Metal Gear Solid* (PC-Cdrom version: Microsoft / Konami 2000), a true masterpiece, at one stage of the game pits the player against a character with psychic powers, Psycho Mantis, who is able to anticipate the player's every move and thus avoid all his attack attempts. The player then receives a suggestion: the only way to keep Mantis from reading the direction that the player gives to his character and to have a chance at winning the duel is to unplug his joypad from port one and to insert it into port two. The first example illustrates the ambitious attempt at extending the game to the experience of the player as a whole (if he does not win, he risks losing his soul and having his real-life body controlled by the bad guys from the other dimension). The second constitutes a rather unsettling experience, that jolts the player back to his own world, and forces him, to interact with the hardware interface, precisely the one that ergonomics experts are doomed to disappear, or at least to become as transparent as possible.

Going back to environment design, the claim that the virtual environment is built by game designers to comply with players' requirements and expectations may be easily verified. Real-Time Strategy games (RTS) are based on conquering landscapes: by controlling his forces (characters, vehicles, planes, etc.), and in real time – that is, without having to wait for the opponent's turn –, the player must beat the enemy army deployed in the theatre of operations, the area to be conquered. To achieve conquest, he must destroy the enemy base while protecting his own. The wide availability of the internet and online gaming makes it virtually obligatory for any new game of this kind to be playable in the multiplayer mode, that is against one or more human adversaries linked through the Web. Multiplayer games are played in environment that are built on purpose by game designers. Of the two types of landscape are compared, that is the single- and multiplayer ones, they are revealed as profoundly different, despite their apparent similarities. As Professor Espen Aarseth points out in his analysis of the game *Myth* (Take2 / Bungie 1997): "The single player landscapes are asymmetrical, often linear, with one path through it, and with 'evil' troops placed in ambush along the way". The design of the environment linearizes the options for movements, forcing the

player to follow a predetermined path, usually through a series of important places. "The multiplayer landscapes on the other hand, are symmetrical, open and usually arranged around a central point" (Aarseth 2000).

The single player experience is therefore dominated by the linearity of a storyline, while multiplayer games offer no storyline, merely a neutral environment suitable to all players. In the first case, the environment is part of the challenge, of the difficulties awaiting the player, that go well beyond the presence of the enemy to include impossible-to-cross rivers and well-defended bridges, impenetrable forests and canyons most suited for ambushes. In multiplayer game, the environment is turned into a background and support for the challenge between human players, and must therefore be, at least at the beginning of the game, as neutral as possible to all of them.

This in fact was the case a few years back. However, the AI controlling enemy troops has very rapidly evolved, and today's products only very rarely display "stupid" behaviours on the part of computer-controlled units. The role of game environments therefore changes, and linear landscapes for single-player games are no longer necessary. Their main purpose was hiding the too-simple nature of computer-controlled units, while at present the question lies into the need to create an environment – a game level – that may offer multiple potential chances that both sides – computer- and player-controlled characters – may turn to their advantage. It is certainly not casual that two of the biggest successes of the videogame industry of the past few years have been *Unreal Tournament* (Gt Interactive / Epic Games, 1999) and *Quake III* (Activision / id Software, 1999), which single-handedly introduced an innovative gaming style, developed through the possibilities offered by the Internet. Briefly, these games were designed to be primarily a multiplayer experience, but they also may be used in single-player mode, against very "intelligent" computer-controlled opponents – known as BOTs – that act in levels that are often very articulate while remaining extremely restricted. The player is no longer pitted against the environment, but rather against smarter adversaries in a neutral but resourcefully-designed environment.

11. Environmental storytelling

The design of virtual environments is particularly important in those cases in which the spaces where the player-controlled character acts contribute with a number of elements to the development of the story, thus enriching the player's journey with the infinite details that a book or a movie, due to lack of space and time, would be forced to eliminate. An extraordinary example of this instance is *Half Life* (Sierra / Valve, 1998), a game in which the player controls a technician working in an underground desert testing facility where top secret experiments are carried out on hazardous materials.

Something goes wrong, and after a number of sudden explosions, the player must quickly reach the facility's exit to save himself. In order to do so, he must go through a number of rooms and corridors where he witnesses various events that provide him with clues about what has happened. There are tremors and crumbling walls, flickering neon lights, broken-down doors and the bodies of dead scientists, blood spattered on the walls and, judging by the blood-curdling screams heard, some of them must still be alive, somewhere. All this points to something really serious, and not a mere explosion, having happened: the presence of someone – or something – unknown is felt around the base.

In this instance, the virtual environment is not considered as a background against which the characters move, but plays an active role within the story, as well as being a support for the player's actions.

Don Carson, a freelance designer and conceptual illustrator who for the past 17 years has worked as a designer for many theme parks, including Walt Disney World Florida and Disneyland California, explained this concept with an effective parallelism between theme parks and virtual worlds. According to Carson, both experiences have the same objective: to bring people into the man-made worlds and keep them immersed and entertained. Also, the ways in which this objective is reached are the same: to resort to the narration of a story through the experience of travelling through a real or virtual space, and in particular by giving the visitor (in our case, the videogame player) the chance to choose his path and the actions to take. He uses the term "environmental storytelling" (from theme-park slang), that is, when "the story element is infused into the physical space a guest walk or rides through; (…) in many respects, it is the physical space that does much of the work of conveying the story the designers are trying to tell". The same principle is resorted to *en masse* by successful game designers, together with other gimmicks that Carson splendidly illustrates in his article, "Environmental Storytelling: Creating Immersive 3D Worlds Using Lessons Learned from the Theme Park Industry" (Carson 2000).

12. The Bottom Line

This article began by a definition of the term "realism" as applied to virtual interactive entertainment environments. Realism in videogames does not mean to provide a copy of the physical world, which rather serves as an inspiration and point of departure for the construction of virtual worlds designed to deal with the expectations of its future users, which in turn are based on their own experiences of the physical world. Just as painting and sculpture, the construction of virtual entertainment environments has little to do with the slavish reproduction of reality, being rather subjected to the

perceptive, representative and executive capabilities of human beings. Moreover, it is important to point out that all virtual worlds are built with specific aims in mind (educative, clinical, entertaining, training, etc.), and it is the aim of a programme that serves as its underlying framework.

13. References:

- Aarseth, Espen (2000). Allegories of Space: The Question of Spatiality in Computer Games. In Eskelinen, Markku and Koskimaa Raine (Eds.) Cybertext Yearbook 2000, University of Jyväskylä.
- Carson, Don (2000). Environmental Storytelling: Creating Immersive 3d Worlds Using Lessons Learned from the Theme Park Industry. Published online in Gama Network's Gamasutra.com, March 01, 2000. Printable version at: http://www.gamasutra.com/features/20000301/carson_pfv.htm
- Carson, Don (2000). Environmental Storytelling, Part II: Bringing Theme Park Environment Design Techniques to the Virtual World. Published online in Gama Network's Gamasutra.com, April 07, 2000. Printable version at: http://www.gamasutra.com/features/20000405/carson_pfv.htm
- Combatsim.com (1997). An Interview with the F22: ADF and TAW Team.

Available at:

Part One: http://www.combatsim.com/archive/htm/htm_arc1/did1.htm,
Part Two: http://www.combatsim.com/archive/htm/htm_arc1/did2_int.htm,
Part Three: http://www.combatsim.com/archive/htm/htm_arc1/did2_int2.htm .

- Combatsim.com (1997). An Interview with Simon Kershaw, Producer for F22 ADF. Available at: http://www.combatsim.com/archive/htm/htm_arc1/did2.htm .
- Hecker, Chris (2000). Physics in Computer Games. Communications of the ACM, July 2000/Vol. 43, No. 7, 35-39.
- Määttä, Aki (2002). Game Developer Conference 2002: Realistic Level Design for *Max Payne*. Published online in Gama Network's Gamasutra.com, May 08, 2002. Printable version at: http://www.gamasutra.com/features/20020508/maatta_pfv.htm
- Norman, Donald (1988). The Psychology of Everyday Things. Basic Books, New York.
- Poole, Steven (2000). Trigger Happy – The Inner Life of Videogames. Fourth Estate, London.

Rouse III, Richard (2001). Game Desing: Theory and Practice, Chapter Seven: The Elements of Gameplay. Published online in Gama Network's Gamasutra.com, June 27, 2001. Printable version at:

http://www.gamasutra.com/features/20000301/carson_pfv.htm

http://www.gamasutra.com/features/20010627/rouse_pfv.htm

- Silvestri, Stefano (1997). F-22 Air Dominance Fighter: Che la Guerra Abbia Inizio. The Games Machine, 102, 36-41.

14. Games cited:

- *Alien vs. Predator* (Fox Interactive / Rebellion 1999)
- *Alien vs. Predator 2* (Sierra / Monolith 2002)
- *America's Army: Operation* (U.S. Army 2002)
- *F1 Racing Championship* (Ubisoft 2001).
- *F1 Racing Simulation* (Ubisoft 1997)
- *F-22 Air Dominance Fighter* (Ocean / DID, 1997)
- *Gran Turismo* (Sony / Polyphony Digital Entertainment, 1998).
- *Grand Prix Legends* (Sierra / Papyrus 1998)
- *Half Life* (Sierra / Valve, 1998)
- *iF-22* (Interactive Magic, 1997)
- *Max Payne* (God Games / Remedy 2001)
- *Metal Gear Solid* (PC-Cdrom version: Microsoft / Konami 2000)
- *Microsoft's Flight Simulator* (Microsoft, 2002: latest version)
- *Monaco Grand Prix Racing Simulation 2* (Ubisoft 1998)
- *Myth* (Take2 / Bungie 1997)
- *Need for Speed – Porsche 2000* (Electronic Arts, 2000)
- *Quake II* (Activision / idSoftware 1997)
- *Quake III* (Activision / id Software, 1999)
- *The Nomad Soul* (Eidos / Quantic Dream 1999)
- *Toca Touring Car Championship* (Codemasters 1997)
- *U.S. Air Force* (Electronic Arts / Jane's Combat Simulations, 1999)
- *Unreal Tournament* (Gt Interactive / Epic Games, 1999)