

11.1 constructing a game

Since ancient times, games have been an essential part of culture. According to Huizinga, game playing is not merely meant for entertainment, but is (one way or another) fundamental for our survival:

... in the game we confront a function of the living creature which cannot be determined either biologically or logically ...

As a terminological aside, in the english language there is a distinction between *play* and *game*. The difference seems to be primarily that a game is subject to additional rules, which determine the success in the game, whereas play seems to be less constrained and more phantasy-driven. In addition, games seem to have a competitive element, a challenge, where you may either win or lose.

Anyhow, as observed in Semiotics:

visual culture

games are an increasingly important element in our visual culture.

Nowadays, with rapid advances in technology, the game industry is said to be overtaking both the film industry as well as the software industry, in terms of total budget. This situation has a somewhat paradoxical effect on game development, however. On the one hand, it is more easy to develop a game, considering the great variety of tools and technologies that are available. On the other hand, to make a successful game has become more difficult, since the competition is much more harsh and does not allow for amateurism. Nevertheless, game development is worthwhile in itself, even if it does not lead to economic success. As a computer science or multimedia student, you should at least be familiar with the process of game development, and gain experience with (a selection of) available game technology.

game engine component(s) Game development covers a wide range of activities, and although preferably done in a team, requires a correspondingly wide set of skills, artistic as well as technical. When speaking about *game programming* it is important to make a distinction between:

- game play programming
- game engine programming

Game play programming, usually, consists of scripting the behavior of characters or elements of a game scene, using a scripting language such as, for example, Lua¹. It may considered to be part of game design, and certainly requires an artistic approach.

In contrast, game engine programming is a much more technical activity and concerns the development of functional components of a *game engine*, which according to Ultimate encompass:

game engine component(s)

- rendering system – 2D/3D graphics

¹www.lua.org

- input system – user interaction
- sound system – ambient and re-active
- physics system – for the blockbusters
- animation system – motion of objects and characters
- artificial intelligence system – for real challenge(s)

Although it is possible to build one's own game engine using OpenGL or DirectX, or the XNA² framework built on top of (managed) DirectX, in most cases it is more profitable to use an existing game engine or 3D environment framework, since it provides the developer with a load of already built-in functionality. In section 4.4, a brief comparative overview of game engine(s), that may be used to built virtual worlds, has been given. This overview did not include the flex 2 SDK, nor related flash engines, that may be used for the development of online (flash) games.



1

elements of game design Game development projects are similar to multimedia projects, or for that matter software development projects, in that they may be sub-divided in phases covering the definition of requirements, the actual development, as well as testing and delivery. However, instead of looking at these phases, or trying to explicate the differences between software development and game development projects, it seems to be more worthwhile to look at what characterizes game development, and, as a consequence, what must be the focus of game design.

Eventhough the question *what is a good game?* is rather elusive, it is evident that a game must be fun to play! When creating a game, we may according to GameDesign capture fun in two areas:

fun

- in the general flow of the game experience and
- in the individual moments during a playing session.

which, together, determine the player's unique experience in playing the game.

As a first working definition, GameDesign states:

a game is a series of processes that takes a player to a result.

which covers both the skill aspect of game playing, as well as the fact that a player generally must make a series of decisions. Refining this initial definition, GameDesign proposes to characterize a game as follows:

interactive electronic game

²msdn.microsoft.com/directx/XNA

A game is a play activity comprised of a series of actions and decisions, constrained by rules and the game world, moving towards an end condition. The rules and the game world are delivered by electronic media and controlled by a digital program.

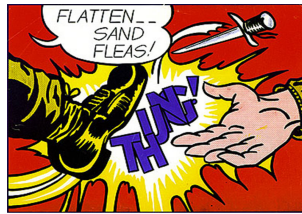
The rules and game world exist to create interesting situations to challenge and oppose the player. The player's actions, his decisions, excitement, and chances, really, his journey, all comprise the "soul of play".

It is the richness of context, the challenge, excitement, and fun of a player's journey, and not simply the attainment of the end condition that determines the success of the game.

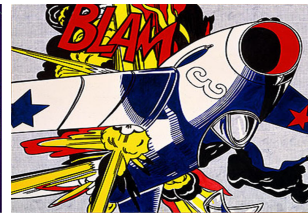
Although this definition could certainly be improved in terms of conciseness and clarity, it does cover the main characteristics of game playing, in particular *context* and *challenge(s)*.



varoom



thung



blam

2

With respect to challenge(s), we may observe that these may take the form of *occasional battle(s)*, as we have called them in the PANORAMA application, discussed in section 5.4. A battle may be characterized by the following conditions:

battle condition(s)

- confrontation on well-established area
- delimited in space/time
- audience/participants who judge victory/loss

It is in this context interesting to note that being mentioned in the *hall of fame*, that is a list of players with the highest scores, seems to be an intrinsic motivation for players to frequently return to the game, either to acquire or maintain a position on that list.

The actual development of a game is a complex process, and, in our experience, for each game unique, since there are many forces influencing a project. Nevertheless, whether you work on your own or in a team, inevitably the following aspects or roles have to be taken in consideration:

design team role(s)

- manager(s) – keep everything together
- producer(s) – maintaining focus
- programmer(s) – solve problem(s)

- tester(s) – control quality
- designer(s) – elaborate idea(s)

In our experience, for example in developing VU Life, that we will discuss in section 11.2, one person may take up several roles, but at some point it becomes clear what the strength(s), and for that matter weaknesses, of each individual participating in the project are.

history of game(s) There are many overviews of the history of (video) games, for example at wikipedia³. One overview that takes an interesting technological perspective is worth mentioning. Algorithms distinguishes between eight phases, where each phase is characterized by one or more unique technical development(s):

history

1. phase i: before space war – hardwired
2. phase ii: spacewar on atari – console with game
3. phase iii: game console and PC – separate game development
4. phase iv: shakedown and consolidation – player code in data files
5. phase v: advent of the game engine – user level design
6. phase vi: the handheld revolution – the GameBoy
7. phase vii: the cellular phenomenon – larger installed user base
8. phase viii: multiplayer games – from MUD to Everquest

For those interested in more details, Algorithms provides as characteristic examples for each phase the following (additional) data: 1) tennis for two William Higinbotham Brookhaven National Labs New York, 1950' 2) Steve Russell, 1961, MIT, spacewar, two player game on Digital PDP-1 3) Atari VCS, Apple II, IBM PC (Dos) 4) Donkeykong, Pacman -> Nintendo 5) Doom -> Valve Halflife 6) Gameboy with well-established collection of game 7) NTT Docomo I-Mode, Samurai Romanesque 8) MUD (1979), MULE (1983), Ultima/Everquest 1600 hours/year.



³en.wikipedia.org/wiki/History_of_video_games

Technical progress does not stop here. Two developments worth mentioning are the rise of Second Life as a 3D immersive interface to the web, and the recent availability of a new 3D engine for flex 2, PaperVision3D⁴, both of which may influence the choice of (online) game platform(s) for the near future. Another development that we may expect for the near future is according to Algorithms: *a truly cinematic gaming experience*.



Screens from Samurai Romanesque.

example(s) – *samurai romanescque*

Samurai Romanesque, available on Japan's NIT DoCoMo packet-switched i-mode network, is an example of a mobile game with a large following. This massive multi-player game is developed by the Japanese game developer Dwango. It runs on the Java 2 platform Micro Edition (J2ME). Players take, as we read in *Samurai* a virtual journey through 15-th century Japan, engage other players in real-time battles, visit historical towns and villages, practice the art of Zen, engage in romances and even can have children. This massive multiplayer role-playing game can accommodate up to half a million players, and is accounted to be a huge success in Japan. *Samurai Romanesque* is an example of a mobile game incorporating features such as position awareness, player history, chatting, and effective graphics. In *Samurai*, it is further explained how the technology with which the game is implemented positions itself in the *battle for mobile cyberspace*.

research direction(s) – *serious games*

Serious games are here to stay. Currently there is for example already a great offer of business management games. When googling on *game*, *business*, and *management*, we find, among many other offerings games to train leadership⁵ (which provides urgent problem situations in a variety of areas, including military and health care applications), and entrepreneurship⁶ (which provides a eight round cycle of sessions instruction how to start a business, get clients, etc., with extensive feedback in the form of reports and comments after each round). A general observation we may make here is, however, that the games we have seen

⁴ papervision3d.org

⁵ www.experiencepoint.com

⁶ www.marketplace-simulation.com

so far primarily focus on functionality and offer at best an efficient interface, which we do not find very appealing from a more artistic perspective.

There are many (more) resources on serious games⁷. To indicate what it is all about, we present a quote from virtual heroes⁸:

Serious games and simulations are poised for a second revolution. Today's children, our workforce and scientists are increasingly playing, learning, and inventing in visually intensive "virtual" environments. In our increasingly experiential economy, immersive educational and training solutions are needed to advance the workforce of tomorrow. Game-based learning and technologies meet this challenge.

However, regardless of the fuss being made, apart from the euphorics there is little attention to determine in a more scientific way what the actual value of the game is in the learning process, and what elements or aspects of the game contribute to the learning experience. To provide such a foundation, we will propose a game reference model in section 12.1, that we have applied to formulate criteria for effective service management games in Serious.

There is a wide choice of technology available for the realization of serious games. For example, in the *climate game* project, we did explore various technologies, including interactive video with flash, as well as the use of the HalfLife2 game engine, with which we gained experience in developing a promotional game for our faculty, VULife. With regard to the use of 3D we may remark that since ancient times a walk in space has served as a mnemonic device, and as such spatial memory may aid in retention and understanding, which might also provide a decisive argument for the use of 3D in a serious game, such as a service management game!

As explained in section 3.4, we found great inspiration for Clima Futura, our climate game, in *Peacemaker*⁹, that provided us with an example of how to translate a serious issue into a turn-based game

From an interview with the makers:

peace maker(s)¹⁰

Q: With the lion's share of strategy games on the market being devoted to ending a conflict through violence, why was it important to you to emphasize the need for a peaceful solution?

A: When we started to work on the project and looked around at other video games, we encountered the notion that war is much more challenging and conflict is essential to engage players. Many people we talked to perceived peacemaking as mere negotiations, where a group of diplomats sit at a table for lengthy discussions and sign agreements. We tried to shed light on what we see as the other side of peacemaking how challenging it is for a leader to gain trust and understanding in the face of constant violence. How difficult it is to execute concessions, while your own population is under

⁷www.cs.vu.nl/~eliens/media/resource-serious.html

⁸www.virtualheroes.com

⁹www.peacemakergame.com

¹⁰seriousgamesource.com/features/feature_071806_peacemaker.php

stress or feeling despair. In a sense, peacemaking can be more complicated, sophisticated and rewarding than war making, and it is a message that we would like to convey to young adults, the future generation of leaders.

In summary, Peacemaker covers both political and social issues, with appealing visuals, not sacrificing the seriousness of the topic. By presenting real-time events using video and (short) text, awareness is created by allowing a choice between the points of view of the various parties involved. Such awareness may lead to political action and will no doubt influence choices, also when it comes to issues of climate change. Essentially, serious games aim at attitude change, the rest follows automatically ...

