Högskolan på Gotland
Passion & vetenskap

Game Save

Game Save Incorporation in Game Design through a MDA Analysis

Kim Aava

Degree Project in Game Design, 15 ECTS Credits
Game Design and Graphics, Spring 2013
Supervisors: Hedda Gunneng, Adam Mayes
Examiner: Steven Bachelder

Abstract

This paper is about save systems and will analyze save functions in various games. Save systems in games are usually blamed for harming the suspension of disbelief as it is an act taking place outside the game and should only be used for its intended purpose: letting the players store their game data, and progress whenever they need.

The game save function developed as rapid leaps of technology were made, with the beginning era of home consoles and home computers allowing players to spend more time playing, creating a demand for longer play sessions in games which lead to a need of methods for storing game data.

The analysis of save systems is carried out in this paper by using *MDA Framework* (*Mechanics*, *Dynamics*, *Aesthetics*), a tool used for documenting game design. This framework can be used to analyze a game from a designer's or player's perspective. By reversing the framework *ADM (*Aesthetics, Dynamics and Mechanics) one can analyze the user experience, which is how the player perceives the game's aesthetics to be, depending on the dynamics and mechanics.

This analysis aims to provide knowledge of how consistency in game design and the save system can be used to create a cohesive game world that facillitates the player's experience and immersion.

Keywords

Consistency, game world, integration, MDA Framework, save systems.

Table of Contents

Glossary

Artificial intelligence (AI)

> A suite of programming techniques that allow a computer to mimic human behavior in certain domains. Video games use AI to provide artificial opponents for players to play against, among other functions (Adams, 2009:633).

First person perspective

First person perspective describes the position of the camera. Players do not usually see the player's character's body; instead the camera is positioned to view the game through the character's eye (Adams, 2009:216).

Game world

> A game world is an artificial universe, an imaginary place in which the events of the game occur. When player enters the magic circle and pretends to be somewhere else, the game world is the place she pretends to be (Adams, 2009:84).

Gameplay

> The challenges presented to a player and the actions the player is permitted to take, both to overcome those challenges and perform other enjoyable activities in the game world (Adams, 2009:640).

Level

> Ordinarily refers to a portion of video game, usually with its own victory condition, that the player must complete before moving on to the next portion. Levels are often, but not always, completed in a prescribed sequence (Adams, 2009:642).

MDA Framework

See section 3.1 for information.

Point and click game

> …point-and-click, in which the player indicates what he wants to do by moving the mouse around the screen (Bates, 2004:6).

Third person perspective

This term does also describe the position of the camera. However, unlike first person perspective, this camera is placed slightly above the character at a fixed distance, which allows the players to see their character (Adams, 2009:216).

1 Introduction

Ernest W. Adams states in *Fundamentals of Game Design, Second Edition* (2009) that "The act of saving a game takes place outside the game world and, as a consequence, saving harms the player's immersion." (Adams, 2009:280). Immersion in this context can be defined in several ways. Gordon Calleja summarizes immersion in *In-Game: From Immersion to Incorporation* (2011).

> …Immersion finds its most frequent use in the context of Digital games. The application of the term, however, varies considerably: it is used to refer to experimental states as diverse as general engagement, perception of realism, addiction, suspension of disbelief, identification with game characters, and more (Calleja, 2011:25).

Today, many games have save features that attempt to encourage suspension of disbelief by using an auto save, or quick save function (see section 4.2.3 and 4.2.7) as it is said to offer the most immersion (Novak, 2011:272). However, according to Kevin Oxland, who writes in *Gameplay and Design,* the saving mechanism is a game design element that is often left out until the last minute (2004:139). This causes problems regarding the overall consistency of the game.

Joshua Mosqueria writes in *Game Design Perspective* (2004) that "Consistency is at the root of immersion." (Laramée, 2004:70) If a game allows a player to perform an action, the player expects to be able to perform that action across the whole game. Mosqueria uses the example of breaking a window. If players are allow to break a certain window, they should be able to break all the windows across the game. Consistency in a game's design is necessary to create a cohesive experience. "In the end, consistency creates an illusion of a real world for the player." (Laramée, 2004:70)

Being consistent with all the game design components, such as a game's save feature can lead to enhancing player experience, and immersion, during play, rather than destroying suspension of disbelief. This paper will use the MDA Framework to examine the saving systems of games, by looking at aesthetics, dynamics and mechanics of the game save features in various ways. Immersion within game save is created when it fulfills the aesthetics and dynamics that are based upon the mechanics of the save systems. A description of the MDA Framework can be found in section 3.1.

2 Purpose

The purpose of this paper is to provide knowledge about the process of saving a game state and the use of various save systems in game design. This knowledge is generated through an MDA analysis. The analysis examines some of the ways the save functions work to be consistent with the game world and what the player perceives, the aesthetics of saving their progress. This is communicated through the dynamics, which are the player’s input in the game and the mechanics, which are the save system of a game. This paper will emphasize designing the game save system as a part of the game world, achieving an internal consistency to keep the player immersed.

3 Methodology

The Intention with this paper is to analyze the save game function of a number of games using the MDA Framework. This paper will emphasize the importance of making the game save function an essential part of the game's design, by analyzing the game save feature of games with the MDA framework. (see section 3.1) Descriptions of various save systems are provided, as well as a historical view of the game save function to give an insight of the origin of saving. The analysis focuses on games that have made an attempt to incorporate the game save function into the game world rather than games that keep it separated. Games that are subjects for the analysis are:

*Last Express (*Brøderbund, Interplay Smoking Car Production 1997)

The Legend of Zelda: Majora's Mask (Nintendo 2000)

Last Window: The Secret of Cape West (Nintendo, Cing 2010)

Resident Evil (Capcom 1996-2011)

Outcast (Infogrames, Appeal 1999)

Grand Theft Auto (BMG Interactive 1997, Rockstar Games 1999-2013)

3.1 MDA Framework

The MDA Framework is an approach for understanding game design and its technical aspects of development. It was introduced during the Game Design and Tuning Workshop at the Game Developers Conference,2001-2004. MDA can be used as a tool to analyze game design, which may be used from a designer's perspective or a player's perspective. The designer creates mechanics that generate dynamics which, in turn generate aesthetics. The player's perspective is the opposite. Reversing the MDA one can look at a player's experience. They experience the game's aesthetics through an interactive dynamics, which emerges from the game's mechanics.

MDA stands for Mechanics, Dynamics and Aesthetics. Mechanics refer to the rules that constrain and guide players, which specify the game as a system. Mechanics are the base components that define the player's action.

Dynamics are based on the relationships among the mechanics in the game and the player's action in the game world. Dynamics evoke the player's response and are created by the player's interaction.

Aesthetics describe the emotional responses that are evoked within the player.

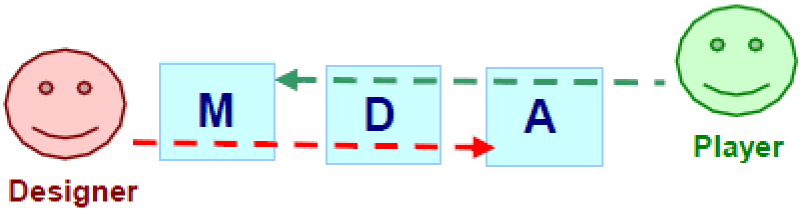


Fig. A. From Robin Hunicke, Marc Leblanc and Robert Zubeck’s paper “MDA: A formal approach to Game Design and Game Research” paper, displaying Aesthetics, Dynamics and Mechanics from the perspectives of a designer and a player.

This paper will be using MDA from a player’s perspective, ADM, to analyze game save design. Where Aesthetics will be determined by how players perceive game save features in relation to the game world that is communicated by dynamics, which is the player's input. Mechanics is the system of game save that the aesthetics and dynamics are emerged from.

4 Background

4.1 Previous research

Game design books tend to offer design paragraphs on game save, and argue that one should allow players to save their game progress. Ernest W. Adams writes in *Fundamentals of Game Design, Second Edition* (2009) reasons for allowing player to save, such as “Allowing the player to leave the game and return to it later.”, “Letting the player recover from disastrous mistakes.” And “Encouraging the player to explore alternate strategies.” (Adams, 2009:279-280) Wayne Imlach writes in *Game Design Perspective* (2002) about game save system and interface, how it may be designed and used in games.

Introduction to Game Development, Second Edition (2010) edited by Steve Rabin, states that players utilize game save systems for three common reasons such as “Stop playing and return later without losing progress.”, “Protect current progress from future failure.” And “Branch their progress to explore alternative choices.” (2010:118) this is similar to the reasons listed by Ernest W. Adams.

Johan Huizinga (1955) used the magic circle in his book *Homo Ludens* to describe the rules of the play, not as the modern way of games, but to describe the temporary world of play, the imaginary world. All play consists of rules that are absolute, those who break the rules break the magic circle (Huizinga, 1955:11). This term is commonly used within game design to describe Imaginary world with a set of arbitrary rules that the players have agreed to act within and that separates them from reality. These rules come to an end when quitting. Ernest W. Adams uses the term magic circle in his book *Fundamentals of Game Design*, where he uses Huizinga's magic circle as an example for the pretended reality (Adams 2009:5).

There are studies and articles that describe the game time, these often contain paragraphs about game save. Jesper Juul (2004) writes in his journal *Introduction to Game Time* about the theory of game time, where it is primarily about linear and measurable time in games. He writes that game save is a manipulation of game time as it allows players to reload previously saved data, which decreases the dramatic tension and also makes the game easier.

Chuck Moran (2010) also writes about game time in the article *Playing with Game Time: Auto-Saves and Undoing Despite the 'Magic Circle'* where he writes that there are other kinds of time in game than linear time flow and uses undoing of actions in games as an example, in which he claims " Linear models of time make it impossible to recognize the time of undoing...". Undoing means that a previously saved file is reloaded, to prevent events such as death, loosing items or other elements that would affect the gameplay in a way the player wants to prevent. One can also look at undoing as rewinding the game time. Moran writes that rewind features create new attractors, replay value increases, as restarting a failed level and death occurs less and does not interrupt play (Moran, 2010).

If a player agrees to a set of rules, in which saving a game state is included as a rule, saving a game is a part of the magic circle and then also part of the pretended reality. However, Chuck Moran writes “Undoing is a practice of play that illustrates how videogaming is temporally imbricated in everyday life, rather than apart from it as many “magic Circle” arguments suggest.” (Moran, 2010) Moran continues by suggesting that the process of trial and error, such as dying and reload cycles is similar to manipulating game time, such as rewinding time is erasing the former failure and similar to reloading the game (Moran, 2010).

Since there are conflicting arguments, when it comes to the subject of the magic circle, in terms of when the rules are broken or upheld within the imaginary bounds, this paper will look to Joshua Mosqueria argument on “Consistency is at the root of immersion.” (Laramée, 2004:70)

4.1.1 Save types

There are several save systems that store the player’s progress. The following list briefly describes the functions of the ones that will be used throughout in this paper.

4.1.2 Password

Goof Troop (Capcom, 1993) for *Super Nintendo Entertainment System (SNES)* uses a password system that offers the player a password after successfully completing a level, which the player can use to start at the next level. This system does not offer the player to save, but lets them start at certain levels when entering the password, without replaying the previous levels.

4.1.3 Save spots

Resident Evil 1-3 (Capcom 1996-1999) uses save spots, which allow the player only to save at certain locations in the game. Players need to interact with a specific object in the environment in order to save as the game does not automatically save the player’s data while players progresses through the game. *The Legend of Zelda: Majora’s Mask* (Nintendo, 2000) for *Nintendo 64* also uses save spot system, but with quick save features, which means that they are only temporary. These save spots are scattered around the game world, but usually occur before a significant event takes place.

4.1.4 Auto save

This system automatically saves game data when reaching certain locations or completing a task in a game. Waine Imlach in *Game Design Perspective edited by* François Dominic Laramée (2002) has named three distinct auto save functions.

- *Time based*, game data are automatically saved every few minutes or at certain intervals of time.
- *Location based*, game data are saved automatically after reaching certain locations.
- *Task based*, game data are saved when completing a greater task or part of the story (Laramée, 2002:188).

Auto save at certain locations, times or task completing can also be referred to as checkpoints (Adams, 2009:282).

4.1.5 Save slots

There are several games that offer save slots, where the player chooses a slot and a name before entering the game for the first time (*The Legend of Zelda*: *Ocarina of Time,* Nintendo, 1998). When saving the game the player is either allowed to choose a slot to save to or it automatically saves to the slot the player chose at the start when saving the game.

4.1.6 Save files

Save files are similar to save slots in the sense that players save their progress by choosing a file. Save file systems do not require the player to create a slot with a name before starting the game. Players can save at multiple files, which makes it possible to reload a previous saved progress, if players choose to use multiple save flies. In *Resident Evil 4* (Capcom, 2004) players can save to different files.

4.1.7 Save anywhere

Save anywhere lets players save at anytime and anywhere during any game state. *The Legend of Zelda: Ocarina of Time* (Nintendo, 1998) has a save anywhere system combined with the save slots system, where players can save whenever they want to a precreated slot.

4.1.8 Quick save

Quick save is much alike the save anywhere function. Quick save lets players save anywhere with a single keystroke, which allows players to reload without exiting gameplay, an example is *The Elder's Scroll: Skyrim* (Bethesda Game Studio, 2011). *New Super Mario bros. Wii* (Nintendo, 2009) uses temporary quick saves, which are deleted once the game is resumed.

It is common to combine several of game save systems that creates a hybrid. Kevin Oxland writes that it's widely used in games today (Oxland, 2004:182).

4.2 History of save game

As a result of home consoles, designers developed methods to let players save their progress as early cartridges could not afford any storage. Initially early game save was represented by a password that allowed players to resume from the last game session by entering a password that could be symbols, texts and digits. This password resumed the game from the beginning of a level. *Goof Troops* (Capcom, 1993) used graphical symbols such as fruits and diamonds. Later, cartridges came with batteries and *RAM* chips that made it possible to store game data on the cartridge instead of entering passwords. Dominic Arsenault mentions in the *System Profile of the Nintendo Entertainment System* that *The Legend of Zelda* (Nintendo, 1986) used a MMC1 chip which allowed players to save their progress to the cartridge, which was designed by developers in order to expand the features of *NES (Nintendo Entertainment System*) (Dominic, 2006).

One of the first desktop computers, *Hewlett Packard HP 9830*, used cassette tapes for storage. Home computers of the late 1970s and early1980s had the ability to use cassette tapes as it was a cheaper alternative to floppy disks (Dubey, 2009:150). In some countries, radio stations would broadcast softwares that were possible to record onto a cassette tape, which then could be loaded into a computer (Dubey, 2009:151).

Jennifer Gipp (2009) writes in *Spotlight on introduction to Computers, Second Edition* the floppy disk was quite fragile and as technology advanced *CD* (compact disk) was introduced as the next type of storage device. Gipp also writes that the first CDs were invented in Japan 1982, with no saving capability, they could only play music (Gipp, 2009: 24). However an early home computer developed by *Sinclair Computers*, later known as *Sinclair research Ltd.* used cassette tapes to save and load data from. Lars Konzack writes that magazines such as *Your Spectrum* or *Sinclair users* came with codes users could type into their ZX Spectrum and save to a cassette tape which was used as external storage for games (Wolf, 2008:203).

Game CD and DVD for home consoles were not used to store save game, instead memory cards were introduced. Leonard Herman writes that "…Playstation was completely CD-driven, gamers could plug-in, for first time ever, external memory cards that would allow them to save game data." (Wolf, 2008:164)

5 Analysis

5.1 Integration of save systems with gameplay

In this section of the analysis, several game’s save functions will be examined through the MDA framework from a player’s perspective, ADM. This paper will look into games that have incorporated game save into the design in a way that makes them a part of the game world. This analysis will be done by examining the aesthetics of the save functions, which will be determined by how a player perceives the dynamics that emerge through the mechanics, as this paper observes.

5.1.1 Last Express

Last Express (Brøderbund, 1997) is a point and click game, where the player interacts with objects and non-playable characters. The game is played through a first person perspective, where the player takes the role of a character that has a certain amount of time to make use of before reaching the end of the game. As game time progresses regardless of player action or inaction, the game can be finished by doing nothing. This is because the game is in real time.

The Setting is 1914 and the game takes place on the *Orient Express.* The train is travelling between *Paris* and *Constantinople.* The player can move around, interact with objects and non-playable characters, which are governed by artificial intelligence and agendas that may change depending on the player’s actions. The player’s actions determine the outcome of the game; meaning that the player might lose before reaching the end destination. To avoid this, the player can turn back time using the clock at the main menu, if the player thinks that they are on the wrong path. The game provides a non-linear story, where players have several actions to choose from. The choice of action will affect the course of the story; players are, however, not told what is the right or wrong path of progress.

Last Express uses an auto save system with no message displayed when progress is saved; which makes players unaware of when saving occurs. The *Last Express* save system allows players to rewind to a previously played game data by using the clock at the main menu. The player might rewind even further back by using a provided map in the main menu, which displays the events the players have passed along their path (see Appendix Fig.1). If failure conditions are met or time reaches the end of the game before the player reach their last destination, the game automatically rewinds back to a previous save data, giving the player a new chance. This would allow the players to perform the correct action in order to successfully reach the last destination.

The Aesthetics of the save function are the real time and time travelling which is provided by the dynamics, persistent time flow and rewinding time. The dynamics of the rewinding relies upon mechanics of the auto save system.

In an interview with Jordan Mechner, the creator of the *Last Express* explains in the book *Game Design: Theory and Practice, Second Edition* that an ordinary save to file system breaks the experience of games (Rouse, 2005:335-336). The game uses a system that keeps track of the player's previous actions on a timeline, which Jordan Mechner refers to as an egg file. "An egg file isn't a saved game; it's essentially a videotape containing not just your latest save point, but also all the points along the way that you didn't stop and save." (Rouse, 2005:336) The egg file is referring to the clock in the main menu screen, which players use to rewind time. The eggs come in six colors, representing different save points. Mechanically, *Last Express* uses an auto save system incorporated with the game's design as it lets the player rewind to a previous auto saved spot.

The story of *Last Express* does not contain the theme of time travelling. Time travelling is a mechanic of the clock in the main menu in which uses auto save function to time travel. Time travelling is an aesthetic that has been generated through the system of auto save. The clock is on the main menu, which means that the player has to exit to another screen in order to use it. The game could be divided into two layers of design, the top layer that is the design of gameplay, where the player is immersed in the game world with real time. The bottom layer that uses auto save system as a support for the time travel aesthetics of the game save. Even though the game save design is not a part of the game world, it is cohesive with the play and creates consistency with the game design, which is the core of immersion.

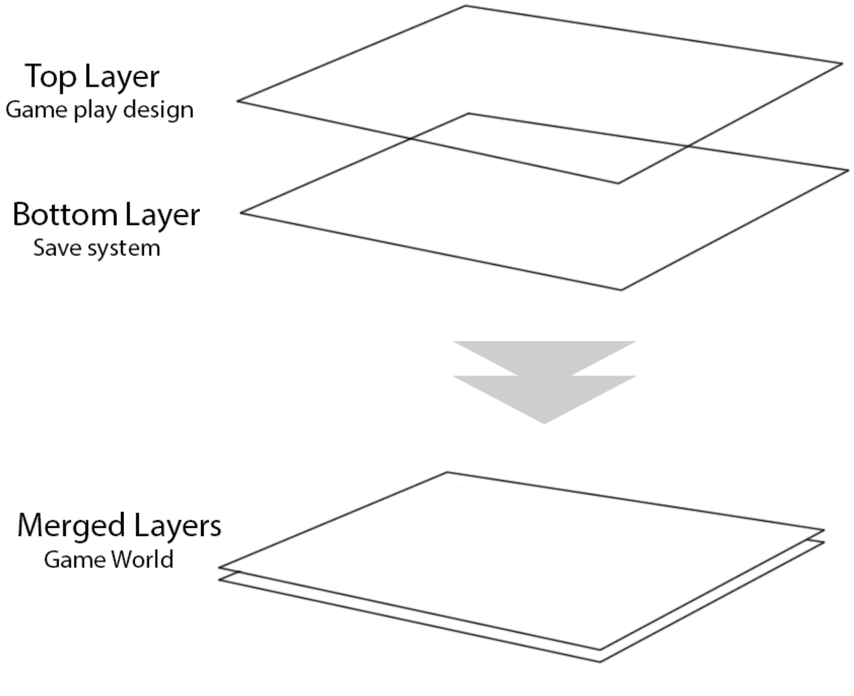


Fig. B. Demonstrates top and bottom layer design hypothesis.

ADM of the save function

Aesthetics: real-time, time travelling

Dynamics: persistent time flow (regardless of player state), rewinding time

Mechanics: auto save

5.1.2 The Legend of Zelda: Majora’s Mask

The Legend of Zelda: Majora’s Mask (Nintendo, 2000) is played from a third person perspective, where the player takes the role of a child who possesses a music instrument, an ocarina, with various abilities. Among them are the abilities of controlling time, teleportation, weather and more. This is done by playing unique songs to each ability. The player can also find other objects such as combat items, funds or other useful objects. The player can interact with non-playable characters, battle or complete puzzles. *Majora’s Mask* imposes an in-game time limitation of 3 days (72 hours) in which players must save the land of *Termina*, before the moon falls to the surface and destroys it. If 3 days come to an end or the player meets failure conditions, the game will end and restart at the beginning of the 3 day-time cycle.

Majora’s Mask uses a combination of save systems that store the player’s data to a save slot, which makes the save system a hybrid. Players can save at various locations by using owl statues, which also serve as a means of teleportation, where players can teleport by playing a specific song on the ocarina. The owl statues are scattered around the game world, the ones that have been found will be displayed on a map that the player has access to through an in-game menu. The owl statues serves as save spots that stores all the player’s data, until the player reaches the end of the 3-day- time-cycle or play a song to rewind time. Owl statues therefore act as temporary save spots which makes them a quick save feature (see Appendix Fig.2).

By rewinding time, the player loses resources such as cleared puzzles, minor objects, funds and non-playable character that will not have any recollection of meeting the player nor the help that has been offered. This will be permanently stored to the save slot. Rewind time is necessary to progress, which also makes the game appear to reset itself back to the beginning of the game, which means the beginning of the 3-day cycle. Major accomplishments will however be stored, such as having retrieved new songs, won essential battles and having found significant objects.

Looking at the save function, the aesthetics being that of time travelling, as the player has to rewind time in order to save and teleport, as the player needs to find these save spots in order to teleport between them. Time travelling is provided by the dynamics of reversing time, which the player does with the dynamics of playing music. Aesthetics of teleporting is provided by the statues which are again triggered by the dynamics of playing music. These aesthetics and dynamics are supported by the mechanics of the save systems.

As mentioned previously, players will lose certain resources when they choose to rewind time; however players can deposit their funds at a bank. Deposited funds will be stored for player to re-collect them after rewinding time. When depositing, a non-playable character marks the player with a seal, as the non-playable character does not have any recollection of the player. The bank acts as an extra inventory bound to a location in the game, an inventory that saves a certain resource, which has the mechanics of a resource stock. The resource stock has the aesthetics of a bank as it is what the player perceives.

As some characters and side quests are only available at certain in-game hours, the save spots system supports the aesthetics of time traveling. The player can, if the day-cycle ends before completion of quests restart the game at the last temporary save spot, which would be the same as the players rewinding themselves back to the beginning by playing the ocarina. The player can redo the quest after completion. These save spots are only temporary which means that if the player turns off the game after reloading once, the game will start from the last time the player rewound time. Players can at will travel back to the beginning of the 3 day-time cycle. When time has run out or failure conditions are met, player will automatically be returned to the beginning of the 3-day cycle. Returning to the beginning of the 3-day cycle is necessary for progression. The player also has ability to slow down time and travel forwards in time by using the ocarina. These save systems are cohesive with the game world as well as game design as it supports the aesthetics of the save function. Additionally, the temporary save spots offer players teleportation to previously found save spots, which add additional functions other than saving.

ADM of the save function

Aesthetics: time travelling, teleporting, bank

Dynamics: reverse time, play music, statues

Mechanics: save spots, quick save, resource stock

5.1.3 Last Window: The Secret of Cape West

Last Window: The Secret of Cape West (Nintendo, 2010) is played on a double screen device; *Nintendo DS*, a handheld console with one touch screen, which in the case of this game the console must be held vertically like a book (See Appendix Fig. 3). Gameplay is divided into chapters such as the chapter of a book. *Last Window: The Secret of Cape West* takes place in *Los Angeles*, 1995 in an apartment building, where the player plays as a former police officer working as a salesman who lives in the apartment. The player must interact with the other residents to reveal crucial information by asking the correct questions and present evidence to solve the mystery surrounding the building's past. The player can interact and use objects that can be found in the surroundings. The player can meet failure conditions by asking non-playable characters the wrong questions, making untrue assumptions or present incorrect evidence. Players have access to a journal and inventory at the touch screen. The journal serves as a note book, where players can write mementos, read about previously met characters, view maps, read a summary of different chapters and also access the save slots. The player can also read the gameplay story as a real book in the game that has the same name as the game, *Last Window*. Each time a chapter is cleared a new chapter is readable in the in-game book that elaborates the story further. Players can also affect the content of the in-game book by their choices during gameplay.

Last Window: The Secret of Cape West uses the save anywhere to a file system. The player can save anywhere expect for certain moments during gameplay such as when interacting with other characters and objects. The player saves by using the journal. The aesthetics of the save function is the novel as the game is an interactive book the player takes part in. That is provided by the dynamics of the journal, which players may use to store their game data and to utilize in additional ways. The journal is available anywhere, except for certain moments of conversation, which supports the mechanic of the save system, save anywhere. As the game is played and handheld as a book and the story is designed as an interactive novel, it supports the journal as a save feature in the game design and is cohesive with the game world. In an interview with Taisuke Kanasaki, in charge of direction and character design of the prequel game *Hotel Dusk: Room 251* at *Cubed3* website, tells that "We chose the style where players hold DS vertically because this matches the style of game itself and its originality - something that has not seen in the other games before."

As players need to exit gameplay in order to access the journal, *Last Window: The Secret of Cape West* can be divided into two layers, similar to *Last Express*. The top layer is where the player is immersed in the game and the story of the novel. The bottom layer is where the player uses the journal to store their progress. Thus, as the handheld console provides two screens, players are still in the game world on the screen with no touch functions (see Appendix Fig. 3). Unlike *Last Express* and *The Legend of Zelda Majora's Mask*, *Last Window: Secret of Cape West's* save system does not have any other purpose than storing the player's data, but is consistent with the additional feature that comes with the journal.

ADM of the save function

Aesthetics: the form of a novel

Dynamics: journal entry

Mechanics: save anywhere

5.1.4 Resident Evil

The main character of *Resident Evil 1* (Capcom, 1996) is a member of the *Alpha* team in *S.T.A.R.S.* (Special Tactics And Rescue Team) on a rescue mission to aid the other team *Bravo* that has gone missing. Team *Alpha* is forced to seek shelter in a nearby mansion after being ambushed in their search for team *Bravo*. The Player needs to obtain documents with clues that help them to solve puzzles within the mansion and uncover the mystery to escape the mansion that is filled with mutated monsters. The Player can find objects, investigate the mansion and battle for survival.

*Resident Evil 1*uses a save spot system to a save file. The Player may save at certain locations by using ink ribbons on typewriters, which are scattered throughout the game world. The players have to carefully choose when to save, as there is a limited number of ink ribbons scattered through the game, like other resources such as ammunition. One cannot use the typewriter without the Ink ribbons which also uses up space in the player's inventory as they also have a limited carrying capacity. Players have to carefully decide whenever it's worth to carry ink ribbons in the inventory, or use that space for other objects, such as ammunition and health packages.

The Aesthetics of the save function is that of writing a document as it is what the player perceive of the game save feature in the game world. Writing a document is provided by the dynamics of using ink ribbons at typewriters which is also a limited object that has to be found. The save feature occupies inventory space that is a mechanics which aids the ink ribbons as a collectible object.

The Save function *of Resident Evil 1* charges players with inventory space as the ink ribbons use inventory space and only a finite number of them can be found, that players need to use as payment to store their game data with. When approaching typewriters, the players are informed that they can save and the game asks whether they want to use the ink ribbon or not (see appendix fig.4). The typewriter is consistent with the game world of collecting documents for clues as well as the resource limitation on other objects.

Resident Evil 2 (Capcom, 1998) and *Resident Evil 3: Nemesis* (Capcom, 1999) use the same save spots system with a limitation of ink ribbons. *Resident Evil 4* (Capcom, 2005) uses a similar system, but without the ink ribbons, players do not need to collect objects in order to save. In *Resident Evil 5* (Capcom, 2009) the save spot system was replaced with auto save.

ADM of the save function

Aesthetics: write documents

Dynamics: typewriter, collect objects,

Mechanics: save spot, inventory space

5.1.5 Outcast

In *Outcast* (Infogrames, 1999) players takes the role of a human character, a former *U.S Navy SEAL* (Sea Air Land Team) that has to escort three scientists during a mission that is set in a parallel world known as *Adelpha*, inhabited by aliens to recover a probe to close a black hole. When arriving in the alien world the player is separated from his team. The Player must find the missing team and help the inhabitants. The Player may interact with the non-playable characters that will treat the player depending on how the player acts towards them. Other actions such as purchase combat equipment, sneak around, completing missions or other task can be done. Enemies will alarm when the player approaches and order nearby foes to attack. The Player may choose to sneak by instead of engaging in combat and can perform certain tasks for regions in order to weaken the enemies.

Outcast uses a save anywhere system to a file. Players can equip an object given to them at the start of the game, called "gaamsavv" and use it to save their progress (see Appendix Fig.5) A non-playable character informs the player in an event in the start of the game, that one can squeeze the gaamsavv to imprint one's essence, in case the player is reverted. By questioning the non-playable character further, the player is also informed that using the gaamsavv near enemies will alert them. The game does not explicitly tell the player that gaamsavv is used to save the game, which keep the immersion high with game world. Players can save anywhere by using this tool, which will make the gaamsavv glow and turn the screen white for a brief moment. If there are enemies near the character during the save progress, enemies will investigate the area.

As the aesthetics are determined by how players perceive the save function, the aesthetics of *Outcast's* saving is the foreign technology, namely the gaamsavv. The player may use the gaamsavv anywhere which is the player's input, the dynamics. Utilizing the gaamsavv creates an additional function, notifying enemies of player's presence.

Outcast has incorporated the foreign technology with the save function, that captures the player's essence. Essence is mentioned during the story of the game that the inhabitants use for many purposes. Save anywhere is represented with an object that supports the aesthetics but also the narrative of the game world.

Saving also features other game design elements, it can cause enemies to locate the player's position. As the player can sneak past enemies, the dynamics is cohesive with the game world. *Outcast's* save system does not only store the player's data, it has consequences for gameplay and the game world. It can be compared to *Resident Evil's* save system, where players is charged by inventory space and a finite amount of saves, by causing disturbance in the gameplay. Immersion is also kept by not explicitly telling the player that the gaamsavv saves the game.

ADM of the save function

Aesthetics: foreign technology

Dynamics: use anywhere**,** affects surroundings

Mechanics: save anywhere, notifying artificial intelligence

5.1.6 Grand Theft Auto

In *Grand Theft Auto* (1997-2013) series the player takes the role of a criminal, and completes missions to get rewards. When players commit crimes the player's wanted level will increase. Wanted level is a scale that measures how much the player is wanted by law.

Grand Theft Auto has used several saving systems throughout the series. *Grand Theft Auto II* (Rockstar games, 1999) uses a save spot system, where player may enter a church with 50,000 dollars in order to save. Players may enter without the requirement but the game will not store game data. In the prequel *Grand Theft Auto I* (BMG Interactive, 1997) the game would only save after finished a city that is made up of different levels.

5.1.6.1 Grand Theft Auto III

Grand Theft Auto III (Rockstar Games, 2001) has its setting in Liberty City, where the player takes on missions, explores, commits crimes, drives cars, uses weaponry and more. Players may receive missions that are connected to the story and side missions. While it is necessary to complete missions to unlock other parts of the city, players can choose to complete the mission at their own pace as the game is non-linear. When the player commits crimes, their *wanted level* will increase. The wanted level is represented by a number of stars on the screen. Upon reaching a certain wanted a level the player will be pursued by various degrees of lawmen.

Grand Theft Auto III uses a save spot system to file, which in the game is referred to as safe houses. The player may save by entering a safe house. There exists a total of three save spots, which can be unlocked through progressing in the story. Save spots are shown as a green house icon on the map, which is available to the player on the screen.

The aesthetics here are the safe house, which the player enters to save their game data that is provided by the dynamics of accessing a building. The mechanics are represented by the save spots, which are the locations of the safe houses.

These save spots also serve as the player's inventory for collectibles and rewards that the player receive from clearing missions. Save spots can also store vehicles, which is similar to *The Legend of Zelda: Majora's Mask* bank that stores the player's funds. The Player needs to drive the vehicles into a garage and then save the game, which is the aesthetics of storing the player's vehicles. The save spots have different capacity for storing vehicles, the last save spot may store up to three vehicles.

ADM of the save function

Aesthetics: safe house, garage

Dynamics: access building,

Mechanics: save spots, resource stock

5.1.6.2 Grand Theft Auto Vice City

Grand Theft Auto Vice City (Rockstar Games, 2002) has its setting in 1986's *Vice City* and uses save spots as *Grand Theft Auto III* and has similar gameplay, however, unlike its predecessor where safe houses are used free of charge, the player needs to purchase their own safe houses. *Grand Theft Auto Vice City* is similar to *Grand Theft Auto II* as the player needs to use funds in order to save. Instead of paying each time as in *Grand Theft Auto II* the player only pays once, which is when they buy the safe house. After Purchase safe houses will then be accessible at all time. The players are able to walk around in safe houses and unlike the prequel game, saving also restore the player's health.

One of the main changes is the floating cassette save icons that did not exist in the predecessor. In Vice City when entering a safe house, the player needs to locate the save icon in order to save. In the prequel, the player only needed to enter safe houses. There are a total of nine safe houses that are available for purchase, not all of them include space to store vehicles.

Grand Theft Auto Vice City uses floating icons to represent save spots, which does not support any immersion. However, the game does have several floating icons for other resources such as weapons, armor, health, funds and upgrades scattered throughout the world. It does then support the game design in which icons are used for other resources as it is consistent with other design elements. The aesthetics of the save function is the cassette icons which the player approaches in order to save. However, if *Grand Theft Auto Vice City* would not have other floating icons to represent resources, the game save would not have any consistency with the rest of the game's design.

Additional *Vice City* takes place in 1986s were save icons are represented by a cassette tape that can be compared to the usage of cassette tapes as storing devices in real life. Other *Grand Theft Auto* games with save icon features have used floppy disks and compact disks and computer disks depending on the year in-game.

ADM of the save function

Aesthetics: icons

Dynamics: approach icons, garage

Mechanics: save spots, resource stock

5.1.6.3 Grand Theft Auto IV

Grand Theft Auto IV (Rockstar games, 2008) is set in 2008s *Liberty City* and has a similar gameplay to *Grand Theft Auto III* and *Vice City* as it's a sequel. *Grand Theft Auto IV* also uses the save spots system to a file. Additional *Grand Theft Auto IV* offers an auto save system that may be turned on or off if not desired. Save spots are still safe houses, however, the player is required to go to a bed in order to save. In these safe houses the player can watch television and use the wardrobe to change clothes. Safe Houses have the appearance of an apartment which holds more functions unlike previous safe houses in *Grand Theft Auto III*. The Garage that stored vehicles has been replaced with yellow lined parking slots outside the safe houses, with the text *"residential parking only"* on the ground. If the player is chased by the police, they will chase the player into the safe house until the player saves their progress at which point the police disappear. Accessing the bed in a safe house will reset any wanted level back to zero.

The aesthetics of this *Grand Theft Auto* game is sleeping that is provided by the dynamics of the player's accessing the bed.

While the mechanics are the same for these *Grand Theft Auto* games, the dynamics and aesthetics separate them. While *Grand Theft Auto III* uses safe houses with limited use, *Grand Theft Auto IV* and *Grand Theft Auto Vice City* offers more functions within the safe house than only saving. In *Grand Theft Auto IV* players accesses a bed in order to save, but it also offers an optional auto save system. By accessing a bed to save it does support the aesthetics as well the game world. Using a bed as a save spot is cohesive with the other gameplay elements of simulating the real life and immerses the player in the game world.

ADM of the save function

Aesthetics: sleep

Dynamics: safe house, access bed, parking slot

Mechanics: save spots, resource stock

5.2 Non-integration of save systems with gameplay

There are some games that incorporate game save as a part of the aesthetics of the game world, however there are more games that do not incorporate game save. This section will analyze briefly some games that do not.

The *Call of Duty* series (2003-2013) uses an auto save system. In the first *Call of Duty* (Activison, 2003) the player could save and reload at any time during gameplay, which means a save anywhere system to file. The sequel *Call of Duty* games use auto save systems, and while it is the same system as *Last Express*, *Call of Duty* does not have any aesthetics nor dynamics for the save game and does not have any other functions that incorporate the save system into the game world. The only purpose the auto save system serves in *Call of Duty* is storing the player's game data automatically.

ADM of the save function in *Call of Duty*

Aesthetics: none

Dynamics: none

Mechanics: auto save

Jeannie Novak in *Game Development Essentials: An Introduction, third edition,* writes that auto saves offer the most immersion during gameplay (Novak, 2011:272). While auto save is purposefully the one that offers the most immersion than other game save options, games like *Call of Duty* do not support any aesthetics of the save functions. Some other examples of games that use auto save systems without aesthetics or any other functions associated with game save are *Max Payne 3* (Rockstar games, 2012), *Diablo III* (Blizzard Entertainment, 2012), *Fallout 3* (Bethesda Game Studio, 2008), *Assassins Creed* (Ubisoft, 2007),*Uncharted: Drake's Fortune* (Sony Computer Entertainment America, 2007), *Super Mario 64* (Nintendo, 1996), *Super Mario World 2: Yoshi's Island* (Nintendo, 1995) The list could be longer, but the point is to demonstrate that auto save is a feature that is commonly used.

As with auto save systems, there are various games that use save anywhere systems without any support of the aesthetics in the game world for the game save or other functions. In *Super Mario Sunshine* (Nintendo, 2002) the player's mission is to collect sunshines by clearing levels. The player is rewarded with sunshine after completing a level; there are a total of 120 sunshines to collect. The player can also collect yellow, red and blue coins. While yellow is to receive health and clear bonus levels, the blue coins is to save the game and purchase sunshines. Red coins exist in certain levels and are collected in order to clear a specific challenge of that level. The player is also asked to save their game after a level is cleared and is also able to save by pressing start to access a menu. While saving after a cleared level or accessing a menu are not a cohesive design with the rest of the game's design, saving by successfully finding a blue coins is consistent with the design of collecting coins. As collecting coins is not the main purpose of the game, it is not cohesive with the game world. This is similar to

Grand Theft Auto Vice City that use save icons, that have consistency throughout the game’s design as other elements of the game uses icons. *Super Mario Sunshine* does not share the same consistency as coins are used to gain health and may be collected to retrieve bonus stars. Blue coins may be used to purchase additional sunshines, it is resource bound to sunshine collecting. However, there are other design elements that are not represented by coins, which make it less cohesive with the game world. Additionally, the save system consists of different save systems, saving by finding blue coins, access a menu and save after a cleared level. Only one of these save systems do have consistency with the resource system of coins.

In *The Legend of Zelda: Ocarina of Time* players can access a menu to save their progress. Often save anywhere systems are accessed through a menu, except from *Outcast* which does incorporate game save by equipping the save feature as an object. An older example of the save anywhere system is featured in *Pokémon Red* (Nintendo, 1996 JP), where the player accesses a menu during game play order to save their game data.

Quick save systems allow players to save instantly and reload at the latest quick save. In *F.E.A.R.* (Vivendi Universal, 2005) players can save by pressing F5 on a keyboard and reload from the quick save by pressing F9 on the keyboard. *New Super Mario bros. Wii* uses a different quick save system, where players can, when viewing the world map, access a menu that offers them to quick save (see Appendix Fig.6). Players select a level on a board map to play, when players reach the middle of the board or the end a permanent save is offered to the players.

Games with save spots tend to create the save locations integrated with the visuals of the game world. For example, *The Legend of Zelda: Majora's Mask* makes use of owl statues that blend into the environment of the game world. *Final Fantasy X (*Sony Computer Entertainment Europe, 2001) and *Eternal Sonata* (Namco Bandai Games, 2007) uses crystal orbs, although neither of them incorporate game save with the actual game design, it is still cohesive with the visual world. There are other examples of save spots that are visually cohesive with the game world is *Baldur's Gate: Dark Alliance* (Interplay Entertainment, 2001) and *Sudeki* (Microsoft Game Studio, 2004), which use books on stands to present their game save locations.

6 Discussion

There seems to be conflicting argument over how game save functionality should be applied in games, while there are those who argue that game save should be designed early into the development of the games and not be left until the end. “The save mechanism is often another element of game design that gets left behind or left until the last minute.”(Oxland, 2004:139) There are others who stress the importance of letting players save to leave the game and live in the real world. “The reason games need a Save feature is so we can grab some time to live our real lives between games.”(Rollings, 2004:114).

Marcin Szymanski writes in *Game Design Perspectives* (2002),

> …when saving becomes an in-game mechanic, such as when the game requires the player to collect potions or other items in order to save, suspension of disbelief is destroyed (Laramée, 2002:112).

According to Szymanskis statement, *Resident evil'*s saving system would destroy the player’s suspension of disbelief as players need to collect objects and place at save spots in order to save. However, from a MDA perspective the aesthetics support players to pick up ink ribbons to enable them to write at the typewriter as collecting documents is a part of the player’s mission. Utilizing the player’s inventory space is consistent with the design of other objects, using space. Consistency is the core of creating suspension of disbelief, which is also referred to as immersion. Marcin Szymanski also writes,

> What if she has no more Save Potions? What if she is too far away from a save spot? Saving the game should be used for its intended purpose: to allow the player to return to the “real world” when they need to (Laramée, 2002:113).

Allowing the player to save certain progress is not always granted. It opposes arguments like “…you simply must allow the player to save his game whenever he wants, wherever he wants, and as many times as he wants” (Bates, 2004:24) which is similar to Szymanski’s argument of allowing the player to return to the real world when they need to. If it is a part of the cohesive design, players may not need to save as it breaks the consistency of the design in the game world, which does not support the suspension of disbelief. As an example, *The Legend of Zelda: Majora’s Mask*, player progressions are saved through temporary and permanent save functions. Some of the player’s progress will disappear, such as minor items and the none-playable character’s recollection of the player’s character, as it is a part of the game design where the player manipulates time. Players are given the opportunity to store their funds, other than that, certain resources will disappear. This is similar to *Grand Theft Auto’s* garages and parking slots, players can store their vehicles if they want to, otherwise vehicles will disappear.

These are arguments of convenience for players to save and return to the game whenever they see fit. But these arguments also create other problems that affect the cohesion of a game. The disruption of game challenges in order to avoid possible penalties, saving and reloading as an act of undoing previous unwanted consequences is arguably damaging to a game's consistency. This paper argues for consistency of a game world with the save components. Outcast is an example that combines Bates argument of allowing players to save anytime (Bates, 2004:24) while maintaining consistency in the game world. It does so by introducing a penalty of attracting nearby enemies to prevent players from abusing saving and reloading.

Last Window: The Secret of Cape West has features that support saving as a part of the game world, but the game still uses common conventions such as the word ‘save’ in the journal, this could be considered not to be a part of the immersive game world. The game save feature can be further integrated into the game world by using bookmarks if different colors in a similar fashion as *Last Express* egg files. Using the word ‘bookmark’ instead of the word ‘save’ in the journal can also further integrate the save feature into the game world. *Outcast* has abandoned such conventions by not mentioning the word ‘save’, players are instead informed that the gaamsavv can imprint one’s essence. *Outcast* is immersing game world and game design, the player is allowed to save anywhere at any time, but is not told that the gaamsavv serves this function during gameplay, only implies it. *Grand Theft Auto* could also avoid this convention by exchanging the word ‘save’ for ‘sleep’ when approaching the bed.

7 Conclusion

While MDA does not look at a game in terms of immersion, it may be used as a tool to examine what players perceive in the game world. The rules by which the players interact and what they perceive is based upon, namely the aesthetics, dynamics and the mechanics.

As Gordon Calleja wrote in *In-Game from immersion to incorporation* (2011) the use of the term immersion varies greatly (Calleja, 2011:25). Save function is stated to break immersion as it is an act that takes the players out of the game world. Consistency is the key word and the core to create immersion. The purpose of this analysis was to provide knowledge about games that has consistency in contrast to games that do not. Consistency can be achieved by a cohesive design through all game components. As seen in many games that are mentioned in this paper, it is possible to incorporate save functions as part of the cohesive game world. Through this cohesion it is possible to make saving a part of the game design and game world, which can be used in several ways to emphasize player experience and create immersive play. Since it is possible to create a complete save system that is cohesive with the game world, there are no reasons why game save should not be an essential part of the game's design or play.

References

Literature

Adams, Ernest W (2009). *Fundamentals of Game design, Second Edition*. California,USA. Pearson Education Inc.

Bates, Bob (2004). *Game Design, second edition*. Thomson Course Technoology PTR. Boston, USA.

Calleja, Gordon (2011). *In-Game from immersion to incorporation.* Massachusetts Institute of Technology, USA.

Dubey, Dr N B (2009). *Office Management: Developing Skills for Smooth Functioning.* Global India Publications Pvt Ltd. New Delhi, India.

Gipp, Jennifer (2009). *Spotlights on Introduction to Computers, second edition.* Course Technology Cengage Learning. Boston, USA.

Huizinga, J (1955). *Homo Ludens: A study of the play element in culture*. Boston: Beacon

Laramée, François Dominic (2002). *Game Design Perspective. Charles River Media Inc. Massachusetts, USA.*

Novak, Jeannie (2011). *Game Development Essentials: An Introduction, third edition.* Cengage Learning, Boston, USA

Oxland, Kevin (2004). *Gameplay and Design*. Pearson Education. Essex, England.

Rabin, Steve (2010). *Introduction to Game Development, second edition*. Course Technology, Cengage Learning. Boston, USA.

Rollings, Andrew. Morris, Dave. (2004) *Game Architecture and Design: A New Edition*. New Riders Publishing. Indiana, USA.

Rouse, Richard (2005). *Game Design: Theory and Practice, second edition.* Wordware Publishing, Inc. Massachusetts. USA

Wolf, Mark J.P (2012). Encyclopedia of Video Games: The Culture, Technology, and Art of Gaming. Volume 1. ABC-CLIO, LLC. California, USA

Wolf, Mark J.P (2008). *The Video Game Explosion: A History from Pong to Playstation and Beyond.* Greenwood Publishing Group, Inc. Westport, USA

Internet

Juul, Jesper (2004). *Introduction to Game Time*. Available at http://www.electronicbookreview.com/thread/firstperson/teleport 20130415

Game Design and Tuning Workshop at the *Game Developers Conference,2001-2004.* Available at http://algorithmancy.8kindsoffun.com/GDC2004/ 20130414

Hunicke, Robin. LeBlanc, Marc. Zubek. *MDA: A Formal Approach to Game Design and Game Research.* Available at http://www.cs.northwestern.edu/~hunicke/MDA.pdf 20130507

Moran , Chuck (2010). *Playing with Game Time: Auto-Saves and Undoing Despite the 'Magic Circle'.* Issue 16 2010:Counterplay. Available at http://sixteen.fibreculturejournal.org/playing-with-game-time-auto-saves-and-undoing-despite-the-magic-circle/ 20130415

Dominic Arsenault (2006). *System Profile of the Nintendo Entertainment System.* Available at http://www.academia.edu/224760/System_Profile_The_Nintendo_Entertainment_System_NES_ 20130423

Riley, Adam. Cubed3, Hotel Dusk: Room 251 interview. Available at http://www.cubed3.com/news/7874/1/-cing-talks-hotel-dusk-another-code-for-wii-and-ds-plus-more.html 20130425

Games

Blizzard Entertainment (2012). *Diablo III*. Microsoft Windows

Bethesda Game Studio (2008). *Fallout 3*. Microsoft Windows

Bethesda Game Studio (2011). *The Elder's Scroll: Skyrim*. Microsoft Windows.

BMG Interactive (1997). *Grand Theft Auto 1.* PlayStation 1

Brøderbund, Interplay, Smoking Car Production (1997). *Last Express*. Microsoft Window

Capcom (1993). *Goof Troop.* Super Nintendo Entertainment System (SNES).

Capcom (1996). *Resident Evil 1*. PlayStation

Capcom (1998). *Resident Evil 2*. PlayStation

Capcom (1999). *Resident Evil 3: Nemesis.* PlayStation

Capcom (2005). *Resident Evil 4*. Nintendo Game Cube

Capcom (2011). *Resident Evil 5*. PlayStation 3

Infogrames, Appeal (1999). *Outcast*. Microsoft Windows

Interplay Entertainment (2001). *Baldur's Gate: Dark Alliance*. PlayStation 2

Microsoft Game Studio, Climax Group (2004). *Sudeki.* Microsoft XboxNamco Bandai Games, Tri-Crescendo (2007). *Eternal Sonata.* Microsoft Xbox 360

Nintendo (2000). *The Legend of Zelda: Majora's Mask.* Nintendo 64

Nintendo (1998). *The Legend of Zelda: Ocarina of Time.* Nintendo 64

Nintendo (2009). *New Super Mario Bros. Wii*. Nintendo Wii

Nintendo, Cing (2010). *Last Window: The Secret of Cape West.* Nintendo DS

Nintendo, Cing (2007). *Hotel Dusk: Room 521*. Nintendo DS

Nintendo, Game Freak (1996 Japan Release). *Pokémon Red.* Game Boy

Nintendo (2002). *Super Mario Sunshine*. Nintendo Game Cube

Nintendo (1995). Super *Mario World 2: Yoshi's Island*. Super Nintendo Entertainment System

Nintendo (1996). *Super Mario 64*. Nintendo 64

Rockstar Games (1999). *Grand Theft Auto 2.* PlayStation 1

Rockstar Games (2001). *Grand Theft Auto III.* PlayStation 2

Rockstar Games (2002). *Grand Theft Auto Vice City.* PlayStation 2

Rockstar Games (2008). *Grand Theft Auto IV*. PlayStation 3

Rockstar Games (2012). *Max Payne 3*. Microsoft Windows

Sony Computer Entertainment America, Naught Dog (2007). *Uncharted: Drake's Fortune*. PlayStation 3

Sony Computer Entertainment Europe, Square (2001). *Final Fantasy X.* PlayStation 2

Ubisoft (2007). *Assassin's Creed.* Microsoft Xbox 360

Vivendi Universal (2005). *F.E.A.R.* Microsoft Windows

Appendix

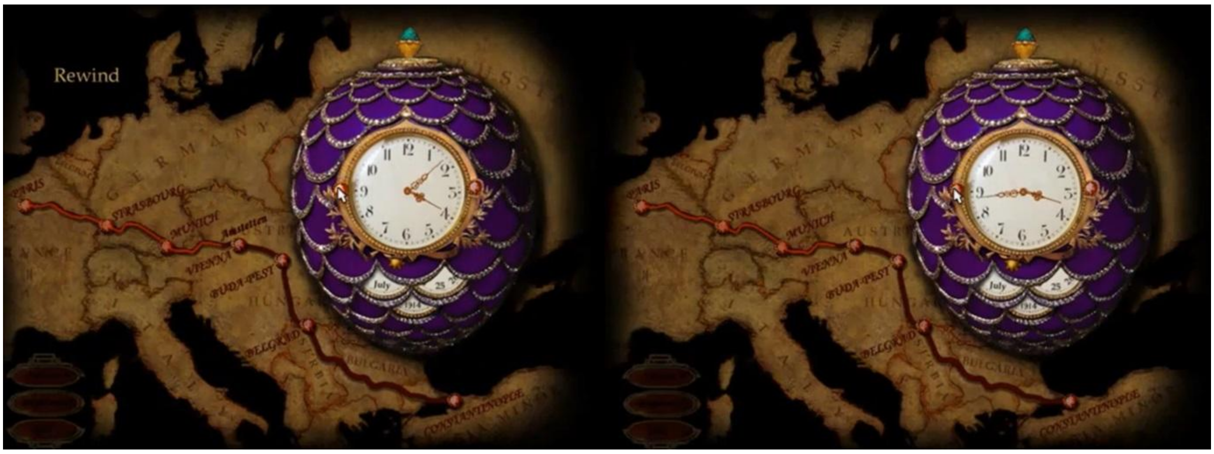
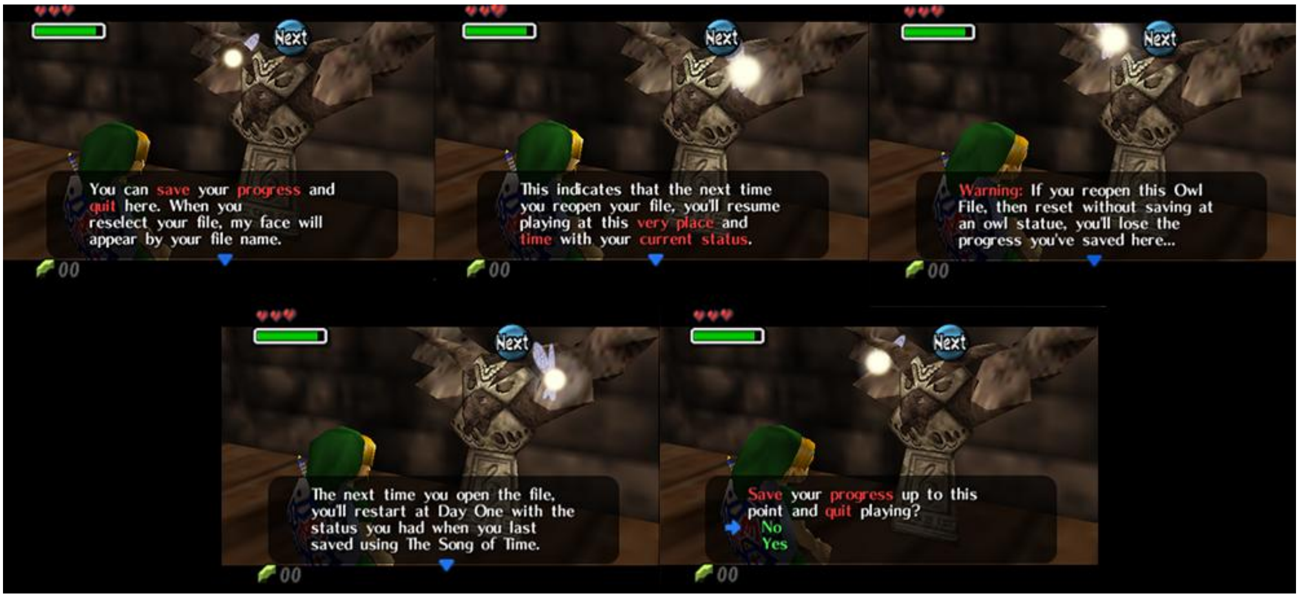


Fig.1: The photos are screen shots from the game Last Express and illustrate the main screen. The clock and the egg file that the player can use to rewind time.



You can save your progress and quit here. When you reselect your file, my face will appear by your file name.

This indicates that the next time you reopen your file, you'll resume playing at this very place and time with your current status.

Warning: If you reopen this Owl File, then reset without saving at an owl statue, you'll lose the progress you've saved here...

The next time you open the file, you'll restart at Day One with the status you had when you last saved using The Song of Time.

Save your progress up to this point and quit playing?
No
Yes

Fig.2: The photos are screen shots from the game The Legend of Zelda. Majora's Mask and displays what owl statues tell players during the first interaction.

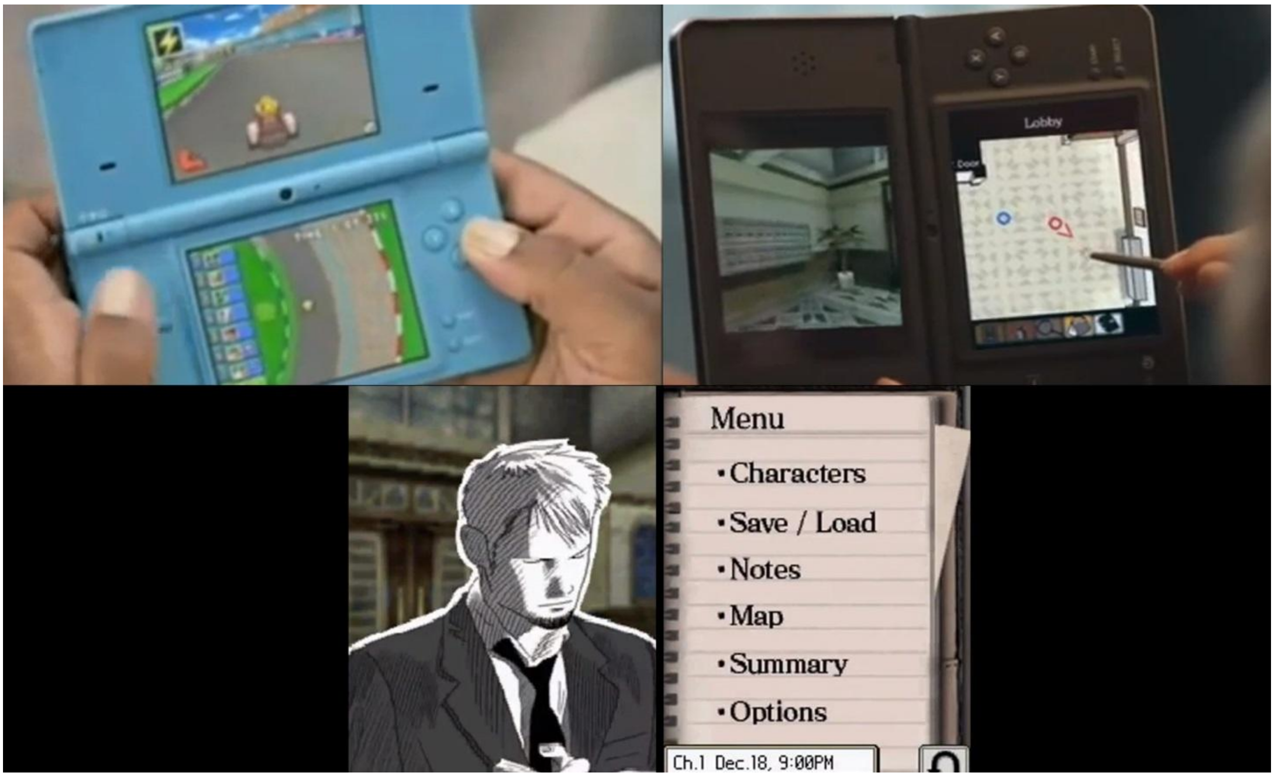
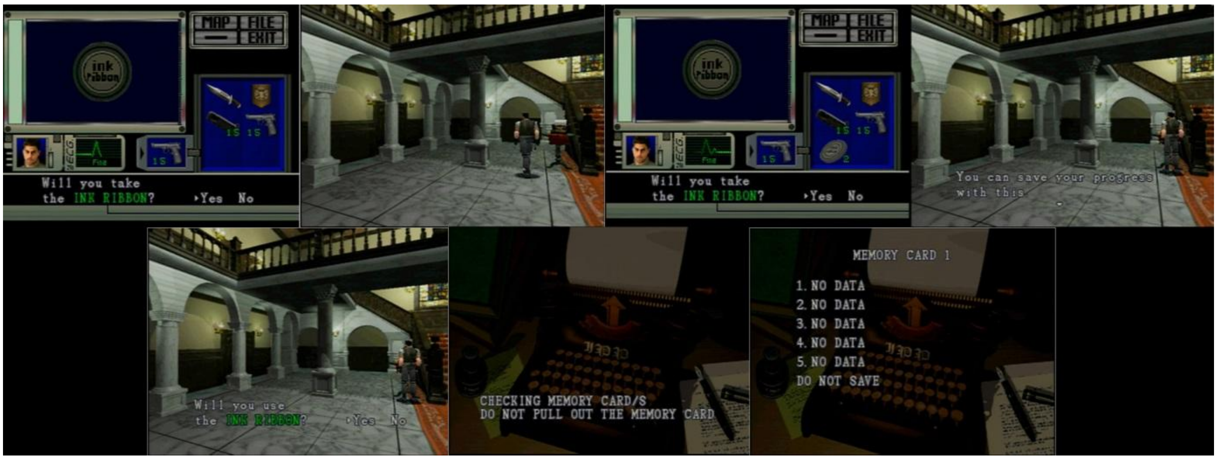


Fig.3: Nintendo DS, handheld console. The photo to the right illustrates how the DS is held when playing Last Window: Secret of Cape West. The photo to the left illustrates how the DS is usually held when playing games and the photo on the bottom display a close up on both screens, where the right side show the journal open and the left side the main character using the journal. The photos are from Nintendo DS official commercials.



Will you take the INK RIBBON? ▸Yes No

Will you take the INK RIBBON? ▸Yes No

You can save your progress with this

Will you use the INK RIBBON? ▸Yes No

CHECKING MEMORY CARD/S
DO NOT PULL OUT THE MEMORY CARD

MEMORY CARD 1
1. NO DATA
2. NO DATA
3. NO DATA
4. NO DATA
5. NO DATA
DO NOT SAVE

Fig.4: The photos are screen shots from Resident Evil and display the inventory with ink ribbons, the typewriter in the hall and the saving process when interacting with the typewriter.

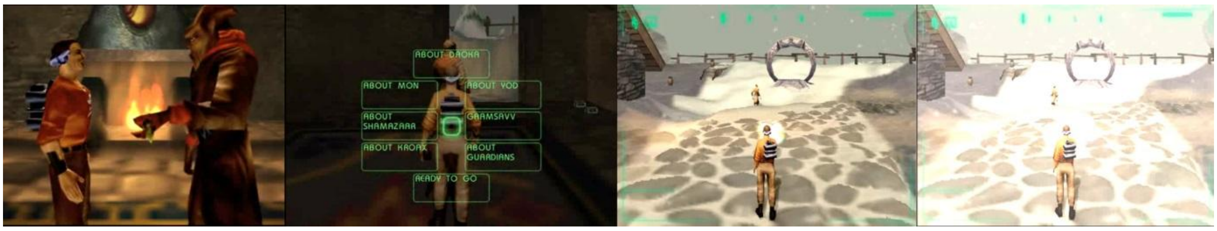


Fig.5: The photos are screen shots from Outcast and display when the player receives the gaamsavv, a dialogue where player can ask about the gaamsavv, when using the gaamsavv and how the screen brightens up when using the gamsaav.

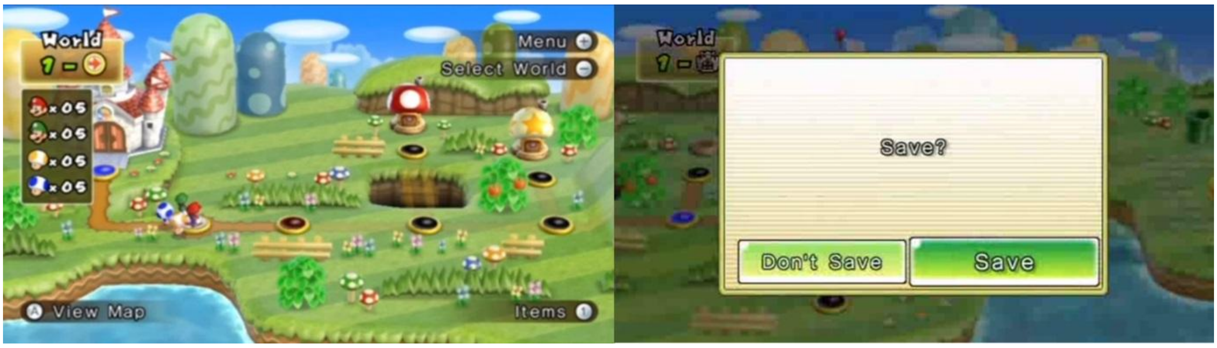


Fig.6: The photos are screen shots from New Super Mario bros. Wii and display the board map and the permanent save.