

**THE TRIBULATIONS OF ADVENTURE GAMES:  
INTEGRATING STORY INTO SIMULATION THROUGH PERFORMANCE**

A Dissertation  
Presented to  
The Academic Faculty

By

Clara Fernández Vara

In Partial Fulfillment  
of the Requirements for the Degree  
Doctor of Philosophy in Digital Media  
in the School of Literature, Communication and Culture

Georgia Institute of Technology  
December 2009

Copyright © Clara Fernández Vara 2009

**THE TRIBULATIONS OF ADVENTURE GAMES:  
INTEGRATING STORY INTO SIMULATION THROUGH PERFORMANCE**

Approved by:

Dr. Janet Murray, Advisor  
School of Literature, Communication and  
Culture  
*Georgia Institute of Technology*

Dr. Jay Bolter  
School of School of Literature,  
Communication and Culture  
*Georgia Institute of Technology*

Dr. Celia Pearce  
School of School of Literature,  
Communication and Culture  
*Georgia Institute of Technology*

Dr. Nick Montfort  
School of Humanities, Arts and Social  
Sciences  
*Massachusetts Institute of Technology*

Dr. Michael Nitsche  
School of School of Literature,  
Communication and Culture  
*Georgia Institute of Technology*

Date Approved: November 13, 2009

## ACKNOWLEDGEMENTS

The author wishes to thank her thesis advisor, Janet Murray, and the rest of the dissertation reading committee: Nick Montfort, Celia Pearce, Jay Bolter and Michael Nitsche. Thanks to Matthew McIntyre at Georgia Tech for all his help with remote paperwork and organization.

Many thanks to the everyone at the Singapore-MIT GAMBIT Game Lab: Philip Tan and Chor Guan Teo, for accommodating the author's schedule to let her work on the dissertation. Thanks to Sara Verrilli for her assistance and for the chocolate. Thanks to best workmates one could have: Jason Beene, Andrew Grant, Marleigh Norton, Abe Stein, Rik Eberhardt, Gene Fierro, Claudia Forero-Sloan, and Mike Rapa.

Also thanks to all the participants in the Work-In-Progress seminar at GAMBIT, who provided constructive feedback during the last year and a half. Special thanks to Jesper Juul, who started the seminar, and those who took part in it: Doris Rusch, Geoffrey Long, William Uricchio, Kristine Jorgensen, Jaroslav Svelch, Pilar Lacasa, Mia Consalvo, Josh Diaz, Kevin Driscoll, Jason Begy and Elliot Pinkus. Also thanks to Henry Jenkins, for his conversations early in the writing process.

Thanks to Manuel Aguirre, Philip Sutton and Bob Shepherd, for their encouragement to study videogames while others thought it was "a waste of time."

Thanks to all my friends and family, for bearing with my absence for so long.

And thanks to Matt, without whose unfaltering support this dissertation would not have happened.

## TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	iii
LIST OF FIGURES.....	x
SUMMARY.....	xi
CHAPTER 1 INTRODUCTION .....	1
1.1. Overview.....	1
1.2. Why Study Adventure Games .....	2
1.3. Thesis Summary.....	4
1.4. Dissertation Outline .....	7
CHAPTER 2 DEFINING ADVENTURE GAME .....	12
2.1. A Crossword at War with a Narrative.....	12
2.1.1. Story-driven .....	14
2.1.2. Puzzle-solving.....	16
2.1.3. Player Character.....	18
2.1.4. Object Manipulation .....	18
2.1.5. Exploration of Space and Action .....	20
2.2. Types of Adventure Games .....	22
2.2.1. Text Adventures/ Interactive Fiction .....	22
2.2.2. Graphical Text Adventures .....	23
2.2.3. Graphic Adventures / Point-and-click .....	23
2.2.4. Adventure Games with Gestural Interfaces .....	24
2.3. The Relationship of Adventure Games to Other Genres .....	25
2.3.1. Interactive Narrative .....	25
2.3.2. Computer Role-Playing Games .....	26
2.3.3. Visual Novels.....	28
2.3.4. Puzzle Games with a Story Setting / Hidden Object .....	30
2.3.5. Escape the Room.....	32
2.4. Conclusion .....	33

CHAPTER 3 STORY AND FICTIONAL WORLDS .....	35
3.1. Reconciling Games and Narrative .....	35
3.2. The Tension between Games and Narrative .....	36
3.3. Narrative Experience .....	40
3.4. Narrative: Story vs Discourse .....	41
3.5. Defining Story.....	43
3.5.1. Causality .....	44
3.5.2. Existents and Events .....	45
3.6. The Fictional World.....	47
3.6.1. Fictional World as Diegesis.....	50
3.6.1.1. Types of Stories in Relation to the Fictional World .....	51
3.7. Conclusion .....	53
 CHAPTER 4 PERFORMANCE IN VIDEOGAMES .....	 54
4.1. Introduction: Dramatic Discourse.....	54
4.2. What is Performance? .....	57
4.2.1. The Basic Qualities of Performance .....	58
4.2.1.1. Time .....	59
4.2.1.2. Objects .....	60
4.2.1.3. Non-productivity.....	61
4.2.1.4. Rules .....	62
4.2.1.5. Performance Spaces.....	64
4.3. Performance Framework for Videogames .....	65
4.3.1. The Theatrical Performance Model .....	68
4.3.1.1. Text .....	69
4.3.1.2. Performance .....	70
4.3.1.3. Mise-en-scène .....	70
4.3.2. Performance in Digital Media.....	71
4.3.2.1. Code .....	72
4.3.2.2. Runtime.....	73
4.3.2.3. Interaction .....	73
4.3.3. Games as Performative Media .....	73
4.3.3.1. Mechanics .....	74
4.3.3.2. Dynamics .....	75
4.3.3.3. Aesthetics.....	76
4.3.4. Implications of the Framework.....	77
4.4. Restoration of Behaviors.....	80
4.4.1. Stories in Performance.....	82
4.4.1.1. Embedded Stories .....	84
4.4.1.2. Emergent Stories.....	85
4.4.1.3. Retelling of Gameplay .....	87
4.5. Conclusion .....	88

CHAPTER 5 ADVENTURE GAMES ARE SIMULATIONS.....	90
5.1. A Story-Game Hybrid.....	90
5.2. Rules .....	91
5.2.1. Rules vs Mechanics.....	92
5.3. Defining Simulation.....	94
5.3.1. The Struggle between Simulation and Story .....	95
5.3.2. Rules-Simulation-Fictional World.....	99
5.3.3. Integrating the Story in the Simulation.....	101
5.4. Abstracting the Fictional World.....	105
5.4.1. Levels of Abstraction.....	107
5.5. The Simulations of Adventure Games.....	110
5.5.1. Different Levels of Abstraction Reflected in the Objects.....	112
5.5.2. Levels of Abstraction in Game Mechanics.....	114
5.6. Conclusion .....	115
 CHAPTER 6 THE GAMENESS OF ADVENTURE GAMES .....	 117
6.1. Puzzles, Stories and Performance.....	117
6.2. Adventure Games as Games of Progression.....	118
6.2.1. Adventure Games are Multisequential.....	120
6.3. Gameplay Rules: Puzzle Solving.....	124
6.3.1. Defining “Puzzle” .....	124
6.4. The Relationship Between Puzzles and Games.....	126
6.4.1. Finding the Solution: Insight Thinking.....	128
6.4.1.1. Designing the Puzzle.....	133
6.4.1.2. Types of Puzzle Logic .....	136
6.4.1.2.1 World use of an object.....	136
6.4.1.2.2 Lateral thinking.....	137
6.4.1.2.3 Designer Puzzles.....	137
6.4.1.2.4 Trial and Error.....	137
6.5. Puzzles in Adventure Games .....	139
6.5.1. Performativity and Puzzles .....	139
6.5.2. The Relationship between Puzzles and Stories.....	141
6.5.3. Types of Puzzles .....	144
6.5.3.1. Navigation Puzzles.....	145
6.5.3.2. State of Affairs.....	146
6.5.3.2.1 Using the right object with the right verb .....	147
6.5.3.2.2 Using the right object with another object/character .....	148
6.5.3.2.3 Manipulating the Environment .....	149
6.5.3.2.4 Timed puzzles .....	149
6.5.3.3. Language Puzzles.....	150
6.5.3.3.1 Find the right word.....	150
6.5.3.3.2 Riddles .....	151
6.5.3.4. Exploration Puzzles .....	151

6.5.3.4.1 Finding information .....	152
6.5.3.4.2 Triggers .....	152
6.5.3.4.3 Hunt the pixel .....	153
6.5.3.5. Mini-games .....	154
6.5.3.5.1 Traditional puzzles .....	154
6.5.3.5.2 Game-inside-a-game .....	155
6.6. Conclusion .....	156
 CHAPTER 7 PERFORMANCE SPACE .....	 157
7.1. The Qualities of Videogame Space.....	157
7.2. Defining Space.....	158
7.2.1. Space is Defined by Practice.....	158
7.2.2. Space is Polyvalent and Transformative.....	159
7.2.3. Space is Performative .....	159
7.3. Space in Videogames .....	161
7.3.1. Affordances of the Designed Space .....	163
7.3.1.1. Navigation.....	163
7.3.1.1.1 Movement .....	163
7.3.1.1.2 Exploration.....	164
7.3.1.2. Participation .....	165
7.3.1.2.1 Contest .....	165
7.3.1.2.2 Puzzle-Solving .....	166
7.3.1.3. Story Setting .....	166
7.3.1.3.1 Navigation.....	167
7.3.1.3.2 Indexical Storytelling.....	169
7.3.1.3.3 Enacted Stories.....	171
7.4. Space in Adventure Games.....	172
7.4.1. Navigation as Gameplay and as Story Device.....	175
7.4.1.1. Mapping .....	176
7.4.1.2. Fragmented Spaces .....	178
7.4.1.3. Labyrinths, Mazes, and Other Navigational Challenges .....	181
7.4.1.3.1 Impossible Spaces.....	184
7.4.2. Inscribed Stories.....	186
7.4.2.1. Descriptions .....	187
7.4.2.2. Documents and Records .....	191
7.5. Conclusion .....	192
 CHAPTER 8 PLAYER CHARACTER .....	 194
8.1. The Player in the Simulation.....	194
8.2. Defining Character.....	194
8.3. What is a Character in a Videogame?.....	197

8.4. The Player Character in Adventure Games .....	200
8.4.1. Functionality of the Player Character .....	201
8.4.1.1. Player Character vs. Avatar .....	202
8.4.1.2. The Problem of Embodiment .....	203
8.4.2. Identity .....	210
8.4.2.1. Player Performance: “Scripting the Interactor” and Restoration of Behavior .....	211
8.4.2.2. Design Elements That Define the Player Character .....	213
8.4.2.2.1 Image .....	214
8.4.2.2.2 Name .....	214
8.4.2.2.3 Gender .....	215
8.4.2.2.4 Genre .....	215
8.4.2.2.5 Actions .....	216
8.4.2.2.6 Dialogue .....	220
8.4.2.3. Types of Player Character in Adventure Games .....	221
8.4.2.3.1 Undefined .....	221
8.4.2.3.2 Generic .....	223
8.4.2.3.3 Defined .....	224
8.4.3. Multiple Player Characters .....	227
8.4.4. Death of the Player Character in Adventure Games .....	234
8.4.4.1. The Player Character Can Die .....	235
8.4.4.2. The Player Character Cannot Die .....	237
8.5. Conclusion .....	239
 CHAPTER 9 NON-PLAYER CHARACTERS .....	 240
9.1. Populating the Simulation .....	240
9.1.1. Simulating the Characters .....	241
9.2. NPC Functionality in the Simulation .....	243
9.2.1. Providers of Information: Conversation Systems .....	243
9.2.1.1. Natural Language Parsing .....	245
9.2.1.2. Choose Your Line .....	247
9.2.1.3. Topic-driven conversation .....	248
9.2.1.4. Canned Conversations .....	250
9.2.2. Part of a Puzzle .....	251
9.2.3. Sidekick .....	253
9.3. NPCs as Animated Characters .....	254
9.3.1. Direct Interaction between Player and Character .....	255
9.3.2. Reaction to Player’s Commands .....	256
9.3.3. Independent Character Actions .....	256
9.3.4. On-cue Actions .....	257
9.4. The Problems with Believable NPCs .....	257
9.5. Conclusion .....	261



CHAPTER 10 TEMPORAL FRAMEWORK OF ADVENTURE GAMES .....	262
10.1. Regulating Time.....	262
10.2. Performance Time.....	263
10.2.1. Event Time.....	263
10.2.2. Set Time .....	265
10.2.3. Symbolic Time.....	265
10.3. Narrative Time.....	266
10.4. Time in Videogames .....	267
10.4.1. Play Time / Real World Time.....	268
10.4.2. Fictional Time / Game World Time.....	268
10.4.3. Fictive Time .....	270
10.5. Temporal Framework for Adventure Games .....	271
10.5.1. Player Time.....	272
10.5.2. Simulation Time.....	272
10.5.3. Story Time .....	273
10.6. Passage of Time as a Design Element .....	275
10.6.1. Real Time.....	275
10.6.2. Cycle-driven Time .....	277
10.6.3. Event-driven Time .....	278
10.7. Conclusion .....	279
CHAPTER 11 CONCLUSION .....	281
11.1. Reconciling Story and Game through Performance .....	281
11.2. Lessons Learned from Adventure Games.....	283
11.2.1. Creation of Fictional Worlds .....	283
11.2.2. Scripting the Interactor through the Design of the Player Character.....	284
11.2.3. Integration of the Story and Puzzles in the Space.....	284
11.2.4. NPC Design .....	285
11.3. Contribution to Game Studies.....	286
REFERENCES.....	288
GAMES AND OTHER SOFTWARE CITED.....	303

## LIST OF FIGURES

Figure 1: Diagram of the different components of the narrative text. ....	45
Figure 2: Performance chart (excerpt). ....	67
Figure 3: The Theatrical Performance Model. ....	69
Figure 4: Parallels between Theatrical Performance and Digital Media ....	72
Figure 5: The Performance Framework for Videogames ....	74
Figure 6: The MDA Framework and its relationship between Designer and Player. ....	76
Figure 7: Stories in relation to the Performance Framework for Videogames. ....	84
Figure 8: Simulation is the intersection between the fictional world and the rules. ....	95
Figure 9: Rules and the fictional world which are not part of the simulation. ....	99
Figure 10: The story in the game is part of the fictional world. ....	101
Figure 11: Events and existents that are not part of the simulation are represented. ....	102
Figure 12: Restoration of behavior generates the story events ....	103
Figure 13: The integration of the story in The 7th Guest ....	104
Figure 14: Level of Abstraction is the overlap between rules and the fictional world. ...	108
Figure 15: Levels of abstraction of Zork vs. Myst. ....	111
Figure 16: Map of the initial area of Zork (excerpt). ....	185
Figure 17: Dichotomy between time inside the media form and outside of it. ....	270
Figure 18: Temporal Framework of Adventure Games. ....	271

## SUMMARY

This dissertation aims at positioning adventure games in game studies, by describing their formal aspects and how they have integrated game design with stories. The adventure game genre includes text adventures (also known as interactive fiction), graphical text adventures, and graphic adventures, also referred to as point-and-click adventure games. Adventure games have been the first videogames to evidence the difficulty of reconciling games and stories, an already controversial topic in game studies.

An adventure game is a simulation, the intersection between the rule system of the game and its fictional world. The simulation becomes a performance space for the player. The simulation establishes how the player can interact with the world of the game. The simulated world integrates a series of concatenated puzzles, which structure the performance of the player. Solving the puzzles thus means advancing in the story of the game.

The integration of the story with the simulation is done through the performance of the player. The game design establishes a specific set of actions necessary to complete both the game and the story, and this set of actions constitutes a behavior that must be restored through performance. The player can also explore the world and its workings, which is necessary to solve the puzzles. By solving the puzzles, the player restores this pre-set behavior.

The simulation in adventure games may not be evident because of a historical shift in the level of abstraction, which determines how the world is implemented in the

game mechanics. Adventure games have increasingly curbed the agency of the player in the world, in order to facilitate completing the story of the game. This move to a less fine-grained interaction has affected different aspects of game design, from reducing the number of possible actions to limiting the interactivity of non-player characters.

The dissertation discusses how adventure games have integrated story with the performance in the simulated world of the game. This integration is further evidenced by how they apply to the four basic elements that bridge story and game design: space, player character, non-player character and time. The qualities of these elements help us understand how the player performs in the simulation, and how that performance is designed.

Analyzing the properties of the simulation in adventure games helps draw comparisons with other videogame genres. The rich history of adventure games can inform the game design of other videogames, particularly in relation to the creation of fictional worlds, strategies to script the interactor, and design of non-player characters.

# CHAPTER 1

## INTRODUCTION

### 1.1.Overview

This dissertation aims at situating the adventure game genre within the field of game studies. The adventure game genre includes text adventures, graphical text adventures, and graphic adventures, also referred to as point-and-click adventure games.

The term “story-game” has been used to refer to adventure games as early as 1985 (Buckles, “The Computer Storygame ‘Adventure’”), and as late as 2004 (Aarseth, “Genre Trouble”). This double nature of adventure games makes them an ideal subject of study. One of the oldest videogame genres, adventure games were the first videogames to deal with the difficulty of reconciling games and stories, a topic that is still controversial in game studies.

The main claim throughout the dissertation is that adventure games are simulations. The basis of every adventure game is a simulated world, which is the environment for both gameplay and story. The player’s performance in the simulated world brings together game and story. As the player interacts with the world within the constraints set by the rules of the game, she is also enacting the events of the story.

The following is a poetics<sup>1</sup> of adventure games, i.e. an overview of their formal properties and different design strategies. There is a caveat, however. The multiplicity of

---

<sup>1</sup> We are using the term ‘poetics’ in the Aristotelian sense, similar to how Laurel and Klastrop apply them to their work. Laurel applies it to Human-Computer Interaction, while Klastrop applies it to virtual worlds.

subcategories that fall under the definition of adventure games are the result of the historical transformation in their interaction design, from text input to mouse input (Fernández Vara, “Shaping Player’s Experience”). This means that the unifying set of properties has a range of variations, which belong to the different types of adventure games.

Some specific types of interaction will be described as being part of the pleasures of adventure games. The aesthetics of adventure games, although not central to the discussion, are relevant to situate adventure games with respect to other videogame genres. This aesthetic assessment also aims at being descriptive, rather than prescriptive. The discussion of the pleasures of adventure games refers specifically to formal aspects of game design, since they may result in different types of player experience.

## **1.2. Why Study Adventure Games**

Adventure games deserve an in-depth study of their game design and of how they incorporate stories with gameplay. They are a genre usually overlooked, even by scholars who approach the study of games from a narratological point of view. Janet Murray, for example, refers to a few adventure games (*Deadline*, *Planetfall* and *Myst*) in *Hamlet in the Holodeck*, but she does not consider them as a separate genre.

Story is inextricable from gameplay in an adventure game. The events of the story take place in a fictional world, which is implemented as a computer program with which the player interacts. Thus, the fictional world becomes a computer simulation, which works in accordance to the rules of the game. Playing the game means making things happen in the fictional world.

The gameplay of adventure games, i.e. the activity of playing the game, is driven

by puzzle-solving, which distinguishes them from other videogames. The puzzles are integrated in the fictional world, so that the player advances in the story as she solves puzzles. Not only that—the player needs to know more about the fictional world and its rules in order to solve the puzzles. It is very difficult to talk about gameplay without referring to the story and vice versa.

Given their longevity as a videogame genre, adventure games have also undergone many changes throughout more than thirty years. The first adventure game, *Adventure*, a.k.a. *Colossal Cave*, was first released to the world in 1976 (Jerz); since then, improvements in computer technology have also transformed both the interaction and game design of adventure games. Successive innovations in the design have generated different types of adventure games: the introduction of graphics was followed by making those graphics interactive; mouse controls have become a staple of adventure games since the early 1990s. The appearance of new types of adventure games has not traditionally rendered pre-existing types obsolete—games of all types are still released, whether they are commercial or produced by fans of the genre.

This continual transformation is both a problem and an advantage. It is a problem because adventure games become a shape-shifting object of study, so that at times it is difficult to make affirmations that would apply to all adventure games. On the other hand, the changes in adventure games have often been attempts to take advantage of technological advances to address specific interaction problems. This provides us with a wide range of strategies that adventure games have followed in order to facilitate the interaction of the player with the world, and to integrate the story with gameplay.

Adventure games also deserve an in-depth analysis in order to dispel a series of

general misconceptions that are associated with them. The game design of adventure games has been dismissed because of its supposed linearity. A closer look at adventure games reveals that, even though they usually have a single solution, the actions and events do not always have to develop in a single order, nor is the player always limited to solving the puzzles that will advance the story.

Another belief that needs clarifying is the commonplace opinion that “adventure games are dead.” The genre topped sales charts during the 1980s and early 1990s, but after the peak success of *Myst*, the avenues for innovation seemed to stall; now adventure games are considered a “niche” market. The genre, however, is far from dead—marketing and sales should not determine the health and quality of videogames of any type. New adventure games are released commercially all over the world, along with a growing community of fans who develop their own games using a wide range of game engines.

### **1.3.Thesis Summary**

What follows summarizes the main argument of the dissertation, a conceptual map of the general thesis. The foundational concept is that playing videogames constitutes a performance. Games in general partake of the common qualities of performance activities: they have rules, they have a special regulation of time, take place in a ritual space, employ objects with a special value within that space, and the activity is non-productive of material goods (Schechner, *Performance Theory* 8-19). By extension, playing videogames is also a performative act.

As the player performs, she is restoring behavior (Schechner, *Between Theater and Anthropology*). Restoration of behavior means performing in the way that is expected



by the activity. This behavior is conditioned by the rules and goals of the game. How the game facilitates the restoration of behavior thus depends on how the game is designed, i.e. what rules and goals have been determined by the game designer(s). The dissertation focuses on how the game design of adventure games shapes the means by which the player restores the behavior that is expected by playing the game.

The design of the game is grounded in the relationship between the rules of the game and the fictional world wherein it is set. Juul bases his discussion of the nature of videogames on the relationship between rules and fiction (Juul, *Half-Real*). The rules define how the player interacts with the game, what the player can or cannot do; in the case of videogames these rules are implemented in the code of the program. Juul recurringly uses the ample term “fiction”; however, he seems to be referring to “fictional worlds” throughout, and that is the term he actually defines. A fictional world is the imaginary world where the game takes place, and will be the term used throughout the thesis.

The fictional world can also host a story, a series of events that unfold as the player plays with the world. The actions of the player cause different events to take place—those events make up the story of the game.

The intersection between rules and the fictional world constitute the simulation. The simulation consists of the aspects of the fictional world that have been implemented in the rules of the game (Juul, *Half-Real* 170); thus the simulation establishes how the world works. The player can interact with the fictional world within the constraints established by the rules. For example, a rule can determine whether the player character can jump or not. Rules also establish certain goals to be attained; the actions to achieve

the goal are again constrained by the rules.

The relationship between rules and the fictional world is the ground on which we can affirm that adventure games are simulations. The fictional world is the basis for most of the interactions with the game—opening and closing doors, turning lights on and off, or talking to characters to obtain information from them. In adventure games, the actions of the player in the fictional world generate the story of the game: as the player performs, the restoration of behavior becomes the story of the game.

What distinguishes adventure games from other videogames is how the behavior is restored. The goal of adventure games is to traverse the game successfully (Montfort, *Twisty Little Passages*), i.e. get to the end of the story. The story is predetermined, and how it must be traversed is also pre-established. Thus, the story sets the ultimate goal of the game. The behavior to be restored becomes very specific, and therefore the game design has to create a system that will allow the player to perform concrete actions that the game expects.

Puzzle-solving is the main game design device that allows the restoration of behavior in adventure games. While the rules of the game establish how the fictional world works, and how the player can interact with it, the puzzles establish the subgoals of the game. Every puzzle is a problem in need of a solution; the goal the puzzle establishes is to solve its problem. In order to traverse the game successfully, the player has to solve a series of puzzles through object manipulation, thus attaining a series of consecutive goals and advancing in the game. Puzzle-solving constitutes the main behavior that must be restored in an adventure game.

What is more important, the puzzles in adventure games are integrated in the

fictional world—the problems are part of the state of affairs of the world: from solving a crime (*Deadline*, *Gabriel Knight: Sins of the Fathers*), finding treasure within an intricate cave (*Adventure*), or defending an innocent person who has been wrongly accused of a crime (*Phoenix Wright: Ace Attorney*). The player must interact with the objects and characters that are part of the fictional world. This integration of the puzzles in the world is what distinguishes adventure games from other puzzle videogames, such as *Bejewelled* or computer versions of Sudoku.

The integration of the puzzles in the fictional world has two important implications for our argument. On the one hand, it means that as the player solves the puzzles, the story advances. The integration also implies that in order to solve the puzzles, the player must learn how the world works. Exploring the world and experimenting in order to learn the rules of the world therefore is fundamental to playing an adventure game.

Throughout the dissertation, we will unpack the main concepts here introduced: performance, restoration of behavior, rules, fictional worlds, simulations, puzzles. These concepts are the pillars to the main thesis: adventure games are simulations, therefore the performance of the player in the simulation is what brings together game and story.

#### **1.4.Dissertation Outline**

The dissertation elaborates on the main argument expounded in 1.3, providing extended definitions of the main terms (story, fictional worlds, performance, restoration of behavior, rules, simulation, puzzle), and discussing them in the context of videogames in general and of adventure games in particular. The first part of the dissertation is devoted to defining these terms. Defining the adventure game in more detail, and how it

differs from other genres is the first step to define the corpus of the thesis—that is the purpose of Chapter 2.

The ensuing chapters define and discuss the terms listed above, incorporating them into the extended discussion of how performing in the simulation can reconcile story and game. Chapter 3 explains the tensions arising from integrating a story into videogames, and expands on how fictional worlds become the stepping stone to bring story and game together.

Chapter 4 deals with how playing videogames is performance, and with how the different aspects of the performance can lead to different types of stories in relation to the game. A performance framework for videogames becomes part of the theoretical foundation for the remaining chapters. The process of “restoring behavior” (Schechner, *Between Theater and Anthropology*) is explained here, and becomes one of the running concepts of the dissertation: the restoration of behavior in adventure games is inescapably linked to the story of the game.

The goal of Chapter 5 is establishing and discussing the main thesis of the dissertation: adventure games are simulations. The implementation of the fictional world of the game as part of the rules of the game is the prime evidence in favor of this argument. The design of the simulation establishes how the player performs as she plays the game, the simulation creates the space and the objects of the performance. The simulation of adventure games has evolved through time, along with the relationship between the game rules and the fictional world. The extent to which the fictional world is implemented by the rules of the game is established by the *level of abstraction* of the game (Juul, “A Certain Level of Abstraction”). Through time, the interaction with

adventure games has become less fine-grained, reducing the number of possible actions in order to facilitate that restoration of behavior.

Chapter 6 will deal with the “gameness” of adventure games, starting with their status of “games of progression” (Juul, *Half-Real*). The gameplay of adventure games is driven by puzzle-solving, structured by a series of puzzles that are integrated in the simulation. Puzzles have only one solution, thus the behavior restored as the puzzles are solved is very specific. There are different types of puzzles, depending on the main activity involved in solving them, such as moving in the space or interacting with the environment.

The second half of the dissertation tackles the main design elements that bridge the fictional world and the rules: space, time, and characters. The virtual space becomes the performance space and its objects, where time is regulated according to the performance, and the player performs vicariously through the player character. These are basic elements of the simulation; an in-depth analysis demonstrates how the bridging of story and game happens at the level of specific design elements. Chapter 7 introduces the simulation of space as a virtual location and its affordances. The design of space in adventure games has evolved from a textually to graphically represented space. Special attention is paid to the concept of narrative spaces, and how adventure games have inscribed stories into the space.

Chapter 8 deals with the player character as the channel of the player’s performance in the game. The chapter reviews the different qualities that establish the identity of the character and describes the types of player character found in adventure games. Three design elements deserve particular attention: sets of actions, dialogue

systems, and whether the player character can die or not. The sets of actions are the groups of verbs available to the player, which define both how the player performs in the simulation and define the player character. Dialogue systems involve the player character and non-player characters; adventure games are quite productive to study different conversation models, since they probably present the widest variety of systems of any genre. Finally, the death of the player character is a design element that has key repercussions on how the player tackles the game and puzzle-solving.

Non-player characters and their functionality as a design element will be dealt with in Chapter 9. The discussion will focus on the problems of designing believable characters as co-performers. Adventure games have dealt with this problem longer than any other genre, and have addressed it with a range of design devices: from a variety of dialogue systems to incorporating self-initiative and reactions to player activity. However, adventure games are still far from finding a design solution to prevent non-player characters from behaving like robots. In fact, it seems that in the last fifteen years, commercial games have all but ceased trying to design non-player characters with complex behaviors, leaving the challenge to the experimentations of interactive fiction (e.g. Emily Short's *Galatea*).

Chapter 10 will deal with the regulation of time in games as performance. After having established a framework to understand how time works for the player, in the story, and in the game, we will describe how the regulation of time becomes a design element in adventure games.

The conclusion in Chapter 11 summarizes the lessons learned about the design of adventure games and the issues arising from their heavy dependence on stories. At that

point, we will be able to suggest possible avenues of innovation, for adventure games as well as other story-driven genres.

## **CHAPTER 2**

### **DEFINING ADVENTURE GAME**

#### **2.1.A Crossword at War with a Narrative**

Graham Nelson provides a succinct definition of the genre (“The Craft of Adventure”): “An adventure game is a crossword at war with a narrative.” Nelson thus encapsulates the tension created by the hybrid nature of an adventure game. On the one hand, it is a game where the player has to solve a series of interrelated puzzles in order to complete it. On the other, the game is inseparable from the story, whose events unfold as the player advances in the game. Although Nelson is talking specifically about text games, where finding the right word is key to advancing in the game, this metaphorical definition helps us understand the difficulty of reconciling a narrative with a game, particularly from the standpoint of the game design.

The phrase “adventure games” refers to a wide variety of games, which usually have in common a story that drives the gameplay. “Adventure” is usually associated with a quest-like structure, similar to the models proposed by Vladimir Propp or Joseph Campbell. If we consider following a quest structure as a way to define adventure, games such as *Final Fantasy VII*, *The Legend of Zelda: Phantom Hourglass* and *Psychonauts*, for instance, may be referred to as “adventure games.” However, the structure of the narrative should not be the defining feature of a videogame genre, since narrative is not a common trait of all games. Rather, videogame genres are defined by how the player can interact with the game: different rule sets (e.g. fighting games vs. simulation games), number of players (single player, simultaneous multiplayer, MMOs) or interface design



(e.g. menu-driven fights like the ones in *Final Fantasy VII* vs. direct manipulation in fighting games such as *Street Fighter II*). *Final Fantasy VII*, *The Legend of Zelda: Phantom Hourglass* and *Psychonauts* may have narratives with a similar quest-like structure. However, all these games belong to different, though related, genres—*Final Fantasy VII* is a Japanese Computer Role-Playing Game, *The Legend of Zelda* is an action-adventure, and *Psychonauts* is a platformer. Referring to quest alone to define the adventure game genre is far too general a concept and would yield a corpus of games beyond the scope of this dissertation. Defining genre based on the structure of the story is also limiting, since it would exclude adventure games that follow the detective story structure, for example.

For the purposes of defining the corpus of this dissertation, adventure games are defined as follows:

Adventure games are story-driven videogames, which encourage exploration and puzzle solving and always have at least one player character. The basic interaction of adventure games is based on object manipulation and spatial navigation. Their challenges usually appear in the form of concatenated puzzles, which are integrated in the fictional world.

The phrase “adventure games” encompasses three different types of games, which have been variously referred to as “text adventures,” “graphic adventures,” or “point-and-click games.” This definition is based on what these games have in common, in order to identify their distinctive features.<sup>2</sup> These features (story-driven, puzzle-solving, player

---

<sup>2</sup> The concept of distinctive features is borrowed from phonology (Jakobson, Fant and Halle). A distinctive feature is a basic unit which can be positive or negative; each sound had a unique pattern of distinctive

character, object manipulation, exploration of space and action) may be found in other genres, but only when all of them are present will we be able to call that game an “adventure game”.

### 2.1.1. Story-driven

For the purposes of defining the object of study, “story” means the events that take place in the fictional world. This definition will be expanded and discussed at length in the next chapter.

Adventure games present strong examples of how a story can drive the game experience. Story-driven games are structured around embedded stories, which are the stories that are pre-set before the player starts the game (Salen and Zimmerman, *Rules of Play* 383-5).<sup>3</sup> Embedded stories are scripted in the game—they establish the events that have happened before the game starts and set the motivations for the player. These stories are also the non-interactive stories within the game (e.g. cut-scenes, object descriptions); they can also determine what the player has to do in order to complete the game. We will expand on embedded stories and their relation to game design in Chapter 4.

Adventure games always have a story inextricable from gameplay—advancing in the game means advancing in the story. Tabletop role-playing games are a good example of how gameplay becomes part of a story—the rule books of *Dungeons and Dragons* include tables and descriptions with different statistics attached to player character attributes, weapons, items, and in-game actions. The numbers and descriptions only acquire meaningful value if these actions, objects, and characters are part of a story. The

---

features Therefore the co-existence of certain sound units defines each phonological sound.

<sup>3</sup> Salen and Zimmerman borrow these terms from Marc LeBlanc’s talk at GDC 1999.

Dungeon Master constructs a story, the events of the story unfold one way or another based on the rules, and the events are then narrated orally by the Dungeon Master and the players. The story gives meaning and structure to the entities and actions of the game.

The key to discern a story-driven game lies in the integration between the manipulable entities and actions in the game and the agents and events in the fictional world. There must be a substantial overlap between what the characters do as agents in the story and as entities as part of the game, as well between the values and properties of the objects in the game and in the fictional world.

Some games may present the stories of the manipulable entities as a written paragraph, illustrations or cut-scenes; however, the actions presented in these narrative passages are not available as types of manipulation in non-story-driven games. For example, in *Castlevania: Symphony of the Night*, the player character speaks before battling the bosses of the game, but those scenes are not interactive; talking is not an action available to the player during gameplay. The story of the game is clearly divided between interactive and non-interactive periods.

Another relevant feature of story-driven games is that the actions the player must do to complete the game are mostly pre-determined, in the form of an “ideal walkthrough.” The player is usually expected to traverse the game following specific steps, or at least to reach certain milestones in a particular manner, so that a concrete state of affairs is reached and the story unfolds along with the gameplay. The goals of the game, as well as character behaviors and object properties, are so strongly tied to the story they can constrain the possible paths and strategies that the player can take. Story-driven games tend to curtail the player’s choice, precisely because the player must

interact within that ideal playthrough. If there are several ways to finish the game, they all have to be predicted by the designer/writer.

The ideal playthrough seems to go against the procedural properties of digital media, or of games themselves, where choice is the foundation of what the player does in the game. This is a design problem, a narrative problem and a technological problem. Multiple paths can take the form of branching stories, which can be problematic if not inefficient (Crawford, *On Game Design*). A multiplicity of storylines requires an author writing them all, and the interactor can usually traverse them relatively fast. The encyclopedic properties of computers (Murray, *Hamlet on the Holodeck* 83-90) facilitate storing different storylines, but we are still at an early stage in coming up with procedurally generated stories—computers still have limited capabilities to generate compelling stories on the fly (e.g. Propp-like story generators). Coherent and engaging procedural story generation will be one of the keys of emergent storytelling, if it is ever possible.

In adventure games, story dictates what the player has to do in order to traverse the game successfully; therefore embedded stories are predominant. This is a characteristic feature that distinguishes adventure games from other videogame genres. The core discussion of this dissertation will help us understand how the story is embedded and how the design helps the player complete the ideal walkthrough.

### **2.1.2. Puzzle-solving**

The “gameness” of adventure games is different from other genres, mostly because of their heavy dependence on the story. There is usually no competition involved, so that physical skills and reflexes are scarcely needed to complete the game. It

is also difficult to list what the “rules” of the game are, and player’s choice is limited by the story of the game. Rather, gameplay in adventure games is presented as a series of challenges, which take the form of puzzles. The story advances as the player solves the puzzles of the game.

A puzzle is a challenge where there is no active agent against which the player is competing (Crawford, *On Game Design*); puzzles usually have at most a few correct solutions that must be figured out (Rollings and Adams). In the case of adventure games, there is normally only one correct sequence of actions that will provide the solution; if there is more than one way to solve the puzzle it has to be designed into the game. Puzzles slow down the pace of gameplay, since hand-eye coordination and quick reflexes are not required most times (unlike “action puzzles” like *Tetris*). These puzzles are usually interwoven, so that by solving a puzzle the player obtains an object or information to solve another one, or a new puzzle or set of puzzles appears.

The puzzles in adventure games are also characterized by their integration in the fictional world. The entities involved in the puzzle are also entities in the world; solving the puzzle means achieving a specific state of affairs in the fictional world. Adventure games are thus story-driven puzzle games, where the story and the puzzles use the same signifying systems. This is clear in the case of text adventures, since the story appears written on the screen, and the player must also type to affect the development of the game. Graphic adventure games (i.e. adventure games with a graphical interface) tell stories with images and text; the characters that appeared in the story are also available for manipulation during gameplay.

### **2.1.3. Player Character**

Since adventure games are story-driven, they are bound to have characters who participate in the events. As we saw, characters in the story can become the entities in the game. Characters usually have anthropomorphic qualities, even if they are animals or machines.

Adventure games always have at least one character that the player controls. The player character is the main entity that provides the point of view and carries out the player's commands. Typically, adventure games have only one player character, although some games feature different characters that the player can control through the game. The relationship between the character and the player ranges from surrogate figure to complete detachment; the interface will very much define where it falls within this spectrum, as we will see in Chapter 8. Text adventures tend to present more generic player characters with no name, and usually characterized by having generic occupations (a student in *The Lurking Horror*, a detective in *Deadline*). The advent of graphics brought about specific player characters more frequently; these characters were usually related to a specific narrative genre and evoked a specific visual aesthetic. Thus we have Bilbo in Middle Earth (*The Hobbit* (1983)), Roger Wilco in the *Space Quest* series, or Guybrush Threepwood as an aspiring pirate in *The Secret of Monkey Island*. The player character defines how the player interacts with the world, since she cannot affect the world directly.

### **2.1.4. Object Manipulation**

The interaction with the world is mediated through the player character, which is the entity that manipulates the objects and deals with the non-player characters. The

player's input is a *command* (Montfort, *Twisty Little Passages* 26), which directs the player character to do something in the world. A typical command, inherited from text adventure games, takes the form of a transitive verb plus an object in the room or in the player character's inventory, so that the most common commands involve object manipulation.

Commands establish a dialogue-like interaction with the game, where each command is replied with a specific output, which indicates whether the action has been successful or not. Each command and its corresponding output constitutes a *cycle* in the interaction (Montfort, *Twisty Little Passages* 25). The cycles have been inherited by adventure games with graphics later on, so that most of the interactions still follow a cycle pattern. Thus the interaction in adventure games is normally not in "real time," but rather structured as a series of cycles. This cycle structure also means that there can only be one input / output at a time, e.g. one cannot walk and talk to someone at the same time, they have to be consecutive actions.

Object manipulation can involve one object (e.g. "pick up lamp), or two objects at the same time, for example "use wire with outlet" (*Broken Sword: The Shadow of the Templars*). It can also involve an object and a character, e.g. "give hot dog to boy" (*Gabriel Knight: Sins of the Fathers*) or "use fake barf with Harold" (*Day of the Tentacle*). Some commands in text adventures are intransitive verbs (i.e. verbs that do not need an object to have complete meaning), such as *pray*, *jump* or *wait*, so that the action refers to the player character itself. Those intransitive verbs have been dropped by graphic adventure games, so that all interactions are based on transitive verbs.

As I argued elsewhere (Fernández Vara, "Shaping Player Experience"), the

command input has evolved through time, from typing a verb + object command, to choosing the verb from a menu and then clicking on the object that is going to be manipulated, to choosing an icon that stands for a verb or several verbs and then the object, to clicking on the object that has a specific verb associated with it. Thus the cycle structure has gone from being explicit (in the text commands) to implicit (by clicking on an object that is associated with a specific action). The evolution of command input along with the general interface design, shows how adventure game developers pursued a direct manipulation model of interaction (Shneiderman), so that the player can participate in the simulation of the fictional world. This has taken the form of gestural interfaces in some current adventure games, where the commands have become equivalent to moving the controller in a specific way, imitating the gesture to perform the required action.

### **2.1.5. Exploration of Space and Action**

The last distinctive feature of the genre is that it encourages exploration of the game space through interaction. This means that part of the gameplay is based on exploring the space and interacting with the objects and characters, in order to learn how the fictional world works. This feature is one of the most overlooked qualities of adventure games, because they are usually assumed to force the player to follow a specific linear path to complete them. Costikyan refers to the playthrough structure of adventure games as “beads on a string”:

[...] small areas where there is some freedom of action until some event occurs, at which point a transition to the next bead is opened. While there is some freedom within the beads, the overall game is a linear progress through the beads. (Costikyan, “Games, Storytelling and Breaking the String” 8)



This conception of adventure games, although true, seems to disregard the importance of exploration of the world, as if “some freedom” did not qualify as a relevant activity. Exploration is not a device to fill time while the player figures out how to get to the next “bead”—exploration is a fundamental part of gameplay. The player must scrutinize the world in order to learn more about it; what the player learns from that exploration is essential to solve the puzzles. Finding all the information in a game can also be a pleasure in itself—talking to all the characters to hear their stories, or finding out all the secrets and “Easter eggs” scattered in the simulation may not be necessary to complete the game and yet they are some of the distinct pleasures of adventure games.

Exploration starts with navigating the space, moving from location to location, and examining all the objects in each place. Rather than contested spaces (Squire and Jenkins), the affordances of the space of adventure games are exploration and problem-solving, so that the source of the challenges mostly consists of the environment itself. The usually slower pace of puzzle-based gameplay encourages careful examination of the environment. Adventure games are one of the quintessential examples of spatial narratives, since solving puzzles means advancing in the story as well as opening up new spaces to explore.

The exploration of the space and objects also leads to exploring the possible actions in the world. Reaching a specific state of affairs requires the player to manipulate the objects in the world, investigating their properties and affordances, to then figure out how those objects can help solve the puzzles at hand.

During their historical evolution, adventure games have reduced the amount of exploration that can take place in them, in order to provide a clearer path to the player.

Too much freedom can be intimidating for some players, who need to be guided through the world; having more leeway also means it is more likely to deviate from the behavior that must be restored.

## **2.2.Types of Adventure Games**

There are different types of adventure games, even if they all share the distinctive features just listed. I have already been using terms like “text adventures” or “graphic adventures”; now it is time to specify what these terms mean. The different types are the result of the historical evolution of the interface (Fernández Vara, “Shaping Player Experience”), where each new type was trying to address different interaction problems supported by better computer technology. The changes in the interface have eventually transformed the gameplay experience substantially; even though the interaction is still cycle-based at the core, the evolution of the interface has affected how the player learns about the world and tackles puzzle-solving.

### **2.2.1. Text Adventures/ Interactive Fiction**

In text adventures, the interface is textual, so that the fictional world is described verbally, and the player has to type all the commands. Each cycle constitutes a part of a conversation between the player and the system of game.<sup>4</sup> Their strength comes from a parser which theoretically can understand natural language. In practice, the parser is rather limited—one of the typical complaints about this type of adventure games is the “word hunting” that takes place when the player knows what has to be done, but cannot

---

<sup>4</sup> In Spanish they are called “aventuras conversacionales” (conversational adventures).

come up with the exact wording.

These games present an abundance of locations, which force the player to draw a map as she advances in the game, given the absence of visual cues. The game that started it all, *Adventure*, belongs to this type, as well as *Zork*, *The Hitchhiker's Guide to the Galaxy*, or *A Mind Forever Voyaging*. The genre continues to thrive today thanks to fans, who have appropriated the dated technology to come up with their own stories. The alternative name for this type is Interactive Fiction, which attempts to emphasize the literary qualities of this type of adventure games, rather than their puzzle-like properties (see Montfort, *Twisty Little Passages* 6-8).

### **2.2.2. Graphical Text Adventures**

Graphical text adventures are a transitional type, because they are the same as text adventures, only there are illustrations while the player still had to type the commands. The graphics usually corresponded to each new location, and were not interactive; they were closer to being illustrations in a storybook. Typically, they did not offer that much extra information about the fictional world, because the graphics were low resolution, so the descriptions of locations and objects remained. *Mystery House* (1980) was the first example of this type, followed by games like *The Hobbit* (1982), *Treasure Island* (1985), and *Arthur: The Quest for Excalibur* (1990).

### **2.2.3. Graphic Adventures / Point-and-click**

This type of adventure game depicts the world, characters and objects graphically; unlike the previous type, the graphics are interactive. The player can control the navigation of the player character, for starters, by using the direction keys (*King's Quest*

*I, Fahrenheit / Indigo Prophecy*) or, more commonly, by clicking on the location he should go to (*Broken Sword: The Shadow of the Templars*). Early graphic adventures such as *Maniac Mansion* kept the command line to ease the transition, though the player could choose the verbs from a short list (instead of resorting to word-hunting) and then select the object on the screen he wanted to apply it to.<sup>5</sup> This type is still very popular; recently released commercial games follow this model, such as *The Longest Journey*.

Most graphic adventure games are referred to as “point-and-click,” since they emphasized the use of the mouse and eschewed keyboard input. Graphic adventures emphasize direct manipulation; *Myst* is a prototypical example of the “point-and-click” adventure game. Following this simplification of the interaction, some of the more recent adventure games skip the verb-choosing altogether, so that clicking on an object triggers a specific action associated with it. The *Sam & Max* episodic games (not to be confused with Lucasarts’ *Sam & Max Hit the Road*) and the recent *Ceville* work this way. For example, clicking on a character triggers a conversation with that character; clicking on an item needed to solve a puzzle automatically adds it to the inventory.

#### **2.2.4. Adventure Games with Gestural Interfaces**

One of the problems of point-and-click interfaces is that they restrict the adventure game genre to home computers. Even though the control scheme is not impossible to translate to consoles, it can be quite burdensome to move the pointer with an analog stick. The console game controller, as well as touch-screens (Nintendo DS) and

---

<sup>5</sup> Legend Entertainment Company presented an interesting hybrid interface, blending graphical text adventures with a point and click interface. Every location was depicted and then described in a paragraph. The player could make up the sentence by clicking on the directions on a compass (to navigate), or choose one of the verbs from a relatively long list and one item from another (written) list.

the motion sensors of the Wii controller, permit new control schemes based on gestures, rather than pointing or selecting actions from a menu. These new controllers have brought about new interfaces for adventure games—they offer a chance to emphasize again the importance of the actions (verbs) in puzzle-solving. Games such as *Trace Memory* (US) / *Another Code* (Europe) and *Indigo Prophecy* (US) / *Fahrenheit* (Europe), are the first examples of this revitalization of the genre in newer platforms. These gestural interfaces seem to be bringing back the importance of having a variety of verbs to the genre, since verbs become physical actions that the player has to perform. However, the interactions in the game using this control scheme so far have been rather gimmicky, and only used in certain puzzles. The key to the interaction seems to be managing to perform the right movement at the right moment, rather than offering a range of possible actions.

### **2.3.The Relationship of Adventure Games to Other Genres**

There is a set of videogame genres that relate to adventure games, and are often identified as such. However, not all of them are relevant to this dissertation, because they do not share all the distinctive features of adventure games listed above. The related genres are Interactive Narrative, Computer Role-Playing Games (CRPGs), Visual Novels, puzzle games with a story setting, Hidden Object games, and “Escape the Room” games.

#### **2.3.1. Interactive Narrative**

The term “interactive narrative” is very general. Meadows refers to it as an “art form,” which encompasses a variety of visual arts and interactivity: “An ‘Interactive

Narrative’ is a narrative form that allows someone other than the author to affect, choose or change the plot” (Meadows 3). Meadows does include studies of diverse videogames in his exploration of interactive narrative, such as *Ultima Online* or *The Sims*; significantly, there are no examples of adventure games included in the wide catalog of artifacts.

The relationship between interactive narrative and adventure games is somewhat tenuous. Although in both the interactor / player can affect the events of the story, and the player may choose different paths (if they are available), the main goal of an adventure game is not to change the story, but rather reconstruct it by following an ideal playthrough.

Interactive narrative is also a term used by practitioners like Chris Crawford, who want to distance themselves from games by proposing a new, still nebulous narrative form (Crawford, *On Interactive Narrative*). This puts adventure games in an interesting position—discounted in game studies because they are too much like a novel (Aarseth, “Genre Trouble” 51), the concept of interactive narrative also seems to exclude them because they are too much like games. Thus, the dissertation abstains from discussing adventure games in terms of interactive narrative; although they are a related form, the focus here is on games rather than multiplicity of possible storylines.

### **2.3.2. Computer Role-Playing Games**

The roots of adventure games come from tabletop role-playing games. Will Crowther, the original designer of *Adventure*, played the pen-and-paper version of *Dungeons and Dragons* (the only one available then), and incorporated some of the combat mechanics into the game. When Don Woods expanded the game, he included

various references to the world of *The Hobbit* and *The Lord of the Rings*, works that were also influential in the development of the *D&D* games (Adams and Rollings). Inspired by *Adventure*, the original programmers of *Zork* created a similar game where the player would explore different dungeons, fight enemies, and also encounter Tolkienesque touches along with references to the hacker culture of MIT (Lebling, Blank and Anderson 58).

Computer Role-Playing Games (CRPGs from now on)<sup>6</sup> lack the actual role-playing of a specific character and the collective storytelling that characterizes the table-top genre. Although players can customize their characters in some games (e.g. *Star Wars: Knights of the Old Republic*), it is also common for the game to determine who the player character is (e.g. *Final Fantasy* series or *Planescape: Torment*). What CRPGs have taken from their paper origins is the rule set, which includes the combat system, and in some cases the worlds that those games relate to. For example, *Planescape: Torment* is based on the Planescape campaign setting of *Advanced Dungeons and Dragons* (Cook et al.).

CRPGs are clearly story-driven, following the narrative structure of the quest: the player character is the hero of the game, and has to achieve a specific goal. What is essentially different is that the gameplay of CRPGs is not based on puzzle-solving (although there can be puzzles), but rather on statistics-based combat and abilities (such as lock-picking, tracking, or casting a specific spell). Advancing in the game not only means advancing in the story—it also means improving the statistics of the player

---

<sup>6</sup> We will refer to table-top role-playing games as RPGs, and their computer version as CRPGs from now on.

character who acquires new experience (in the form of points), as it overcomes every obstacle. These experience points allow the player to make the character stronger / more dexterous / more resilient / more intelligent / wiser / more charismatic and learn new skills. Whether a specific action is successful or not is usually decided through virtual dice throws, so determining whether an arrow shot hits an enemy, where, and how much damage it causes is calculated in milliseconds, rather than the several minutes it may take with the pen and paper version.

Depending on the character class, the game can also be completed in different ways, even if the goal is the same. For example, in *Planescape: Torment* the player can choose to be a wizard, which means his combat is based on spell-casting; he may also be able to heal himself and others. If the player character is a warrior he will fight through and heal using potions. Finally, if the player character has high charisma points, he may be able to talk his way out of trouble.

Thus CRPGs and adventure games, although bearing a family resemblance, are different genres: one is based on statistic-driven actions and leveling up the player character, while the other is based on puzzle-solving.

### **2.3.3. Visual Novels**

Visual novels are a Japanese videogame genre, which are more of an interactive book than a game. The screen usually depicts a location in its background, and different characters that appear in each location. The text appears in a box below, which usually depicts what the player character thinks, or the dialogue between the player character and the character on the screen. Most of these games are actually in first person, so that the player is seeing everything through the eyes of the player character.



The main interaction in these games is actually pressing the button to advance the text—there is a considerable amount of text, and usually very little interaction. At specific points, the player is given a set of choices, which will decide how the story will continue.

Visual novels are closer to illustrated (and maybe animated) *Choose Your Own Adventure* books than games. They are also more likely to be considered interactive narrative, given that they can follow branching structures that produce different stories depending on the player's choices. Visual novels are not adventure games because they do not allow the player to explore the world, there is no puzzle-solving, and there is no object manipulation, but rather only choosing options from a menu (e.g. a line of dialogue, or whether to do something or not).

Visual novels, however, have increasingly incorporated elements from adventure games, such as the exploration of the space, or a set of verbs that the player can use at specific stages of the game. The *Gyakuten Saiban* series, known in western countries as *Phoenix Wright: Ace Attorney*, and *Hotel Dusk: Room 215* are examples of how visual novels have incorporated new gameplay elements, so that some of these visual novels can now be included in the corpus of this dissertation. Both *Phoenix Wright* and *Hotel Dusk* allow the player to navigate from one location to another, as well as examining the space, talking to the characters, picking up objects and using them with other objects. Both games also introduce dialogue systems that are a bit more complex than a branching dialogue tree. *Phoenix Wright: Ace Attorney* has courtroom scenes where the player has not only to choose what to say, but when to press a witness, and when to present evidence that contradicts the statement of the witness. *Hotel Dusk: Room 215* is a detective game,

where the player can also press characters for more information at certain points. The moment to press someone for more information is timed, so that if the player misses the chance, she will not be able to do it afterwards. Both games show their origins by having an inordinate amount of text; there are very long periods of time when the player is just reading through paragraph after paragraph, with no option to skip the text.

The visual novel interface is the basis for the interaction with characters and cut-scenes in a variety of genres, from the surgery simulation of *Trauma Center: Under The Knife*, to role-playing games such as *Final Fantasy VII*. Thus, although prototypical visual novels are different from adventure games, the introduction of exploration, sets of verbs and new dialogue systems allows us to include some of the games in the genre in the corpus of this dissertation.

#### **2.3.4. Puzzle Games with a Story Setting / Hidden Object**

There are certain types of games whose gameplay is based on puzzle-solving, and where solving the puzzles advances the story. *The 7<sup>th</sup> Guest* and its sequel *The 11<sup>th</sup> Hour*, along with more recent games such as *Professor Layton and the Curious Village*, belong to this type, and they are usually referred to as “adventure games.” These games, however, are not part of the corpus of this dissertation, because they do not present all the distinctive features of adventure games. To begin with, they are not story-driven. Solving puzzles lets the player know a bit more of the story, but the puzzles are not integrated in the fictional world. In *The 7<sup>th</sup> Guest*, for example, the player has to set up eight queens in a chess board, so that no queen is in a position to capture another, or re-arrange cans with letters to make up a sentence. These puzzles do not involve the characters of the story, nor the actions to solve them can be considered events in a story. These games are more

similar to a book of puzzles, with passages of a story interspersed between the puzzles, rather than letting the story and the puzzle work along together.

Hidden Object games present a similar case. In this genre, the player has to find a set of given objects in a room within a specific time. The room is usually cluttered, and many of the objects can be camouflaged in the environment. Finding all the objects in a room advances the story of the game, which is usually presented in the form of a cut-scene. *Magic Academy* is a stereotypical example of this genre, since all the gameplay is based on finding hidden objects in different rooms from different rooms of a wizard academy. The origins of this genre can be traced back to the “hunt the pixel” puzzles in adventure games, where specific items had to be found by clicking on an extremely small area of the screen, at times one or two pixels wide, or finding an object that was disguised in the background. In spite of the adventure game origins, we cannot consider these games adventure games for the same reasons as above: the gameplay is not story-driven, because the relationship between the actions of the player and the events of the story are completely separate. In the case of *Magic Academy*, the player is a wizard-in-training who has to find different sets of objects for different people. The objects found cannot be examined or used by the player, nor do they figure in the story of the game, which deals with the mysterious disappearance of the sister of the player character. Finding all the puzzles from room to room only allows the player knowing a bit more of the story. Although the core mechanic is “finding,” the player never looks for the missing sister directly. Also, each level is separate from the next, so that the objects found in one rooms will not help in the next one. The puzzles are thus consecutive (one after another) but not concatenated (the solution of one puzzle is part of another).

### 2.3.5. Escape the Room

“Escape the room” games are another genre that originated from a specific mechanic of adventure games, namely, finding a key to open a door. Their gameplay is based on “lock and key” puzzles, where the player has to find consecutive key-like items or pieces of information, until the final key that allows the player to escape the room is found. A typical sequence can be finding a key to open a box that will reveal a code number that will open a safe. Thus the puzzles are concatenated, although they are all of the same kind.

Escape the Room games are browser games, and thus relatively short—the time it takes to complete them can range from a few minutes to a couple of hours. *Mystery of Time and Space (MOTAS)* seems to be the earliest game of this type (2001), and it is relatively long for this genre; the player has to “escape” from 20 levels, so it can take a couple of hours to finish it. *MOTAS* presents itself as an “online graphic adventure game,” clearly revealing its origins. There is a story introduction: the player character’s memory has been wiped, in order to be able to perceive an alternate reality. The puzzle-solving that ensues is supposed to help build up a report on this alternate reality. The game also includes descriptions of each element as the player clicks on them, another clear trait from its adventure game origins. As in the case of Hidden Object games, however, the relationship between the story and the gameplay is quite weak, since the puzzles are really a mostly unrelated series of events (find a key, open drawer, find another key, etc).

Games such as *Crimson Room* or *Sagrario’s Room Escape* have effaced the story of the game to the point where it is not relevant to the puzzle solving. The player has a

single goal: escape the room. *Crimson Room* still has an introduction where the player character wakes up with a hangover (the amnesia or dizziness of the player character seems a staple element of the genre). *Sagrario's Room*, on the other hand, jumps right into business—the player is not given any motivation to escape the room, nor told who the player character is supposed to be, nor why she is trapped in the room. The room itself usually does not resort to space as a narrative element—there seems to be no pre-history to the space (does anybody live there?). The different components of each puzzle do not seem to have any other *raison d'être* than being part of a puzzle (why is there a locked briefcase on the floor?).

Thus “escape the room” games, although a clear offspring of adventure games, have ended up becoming their own separate genre. They are not story-driven, but rather puzzle-driven, and the story is either an excuse for a series of puzzles or non-existent.

## **2.4.Conclusion**

The goal of this chapter has been to define the corpus of the dissertation. Defining “adventure games” does not only mean establishing the object of our study, it also means identifying the features that distinguish them from other videogames. The label “adventure” is applied to a variety of videogames, but only some of those actually qualify as adventure games. Although there will be references to other types of games through the dissertation, here we have established how adventure games may relate to other genres and, more importantly, how they are different.

The distinctive features of adventure games (story-driven, puzzle-solving, player character, object manipulation, exploration) are the foundations for the main argument of the thesis. The following chapters expand on each of these features; although they are all

interrelated, some sections will be devoted specifically to this elaboration. The next chapter defines “story” in relation to videogames, and how it figures with respect to the fictional world of the game. The evolution of the command input is tackled in Chapter 5, whereas a more nuanced definition of puzzle is given in Chapter 6. The role of spatial exploration is covered in Chapter 7. How the player character shapes the performance of the player in the fictional world will be covered in Chapter 8.

## **CHAPTER 3**

### **STORY AND FICTIONAL WORLDS**

#### **3.1.Reconciling Games and Narrative**

Defining what narrative and stories are is still difficult, even in the case of traditional media forms such as the novel. The basic terms of narratology have been imported from literature to film (see Chatman); however, the application of the same terms to videogames has been proved quite problematic. What “narrative” means in the context of videogames is still a contested topic. The essential nature of games and stories seems incompatible: while games are participatory rule systems with a goal, stories are a series of events where the audience does not have any influence. Nevertheless, videogames have the potential to reconcile both, which may result in new media forms. The hybridity of adventure games looks like a first step towards those forms.

The difficulty to bring together games and stories was an early object of controversy in game studies. Frasca affirms, however, that it was a debate that never took place, and was mainly based on a series of misunderstandings (“Ludologists Love Stories Too”).

The opposing sides of this debate were the so-called ludology and narratology. On the one end, ludology was supposed to be the study of games, and videogames by extension, with the aim of understanding their nature. Some ludologists seemed fervently opposed to the suggestion that games could be a new type of storytelling (e.g. see Eskelinen “Towards Computer Game Studies”). The presumed opponents, the narratologists (or narrativists, as Frasca refers to them (“Ludologists Love Stories Too”))

approached games with the analytical tools provided by fields such as literature and game studies, considering them a new form of narrative. These extreme positions correspond to the perception of people outside the debate—as Frasca argues, neither the ludologists were abjuring narratology (many of them are narratologists by training), nor were narrativists negating the inherent properties of games and trying to assimilate them to literature or film studies. Independently of the ontological state of the debate, it has been useful in framing the problem of applying narrative theory to the study of videogames.

What becomes clear from this controversy is that it is essential to define narrative and story clearly, in order to determine how it relates (or not) to games. The term “narrative,” as we stated before, is ambiguous even in the realm of literary theory, since it can refer both to a series of events and to the way they are told. Clarifying the origin of those events, as well as how those events are “told” in videogames, is the goal of the current chapter.

The concept of the fictional world is useful to trace the source of the story events. The fictional world encompasses the setting where the story takes place and provides the characters that will be part of it. The game can share that environment and involve those characters as well. This environment, as we will see in Chapter 5, is the basis to create the simulation where the player can perform, and where the player generates the events of the story as she plays.

### **3.2.The Tension between Games and Narrative**

The application of narrative theory to games is a controversial method. Studying games as literature presupposes that games are a literary text, obliterating the unique aspects of videogames (Aarseth, *Cybertext* 22-3). Although there are works that straddle



games and literature, such as Interactive Fiction (Buckles, “The Computer Storygame *Adventure*”; Montfort, *Twisty Little Passages* vii), videogames in general are not intended to be literary or even film texts. The debate between ludology and narratology has been based on misunderstandings (Frasca, “Ludologists Love Stories Too”), and some of the arguments sounded more like a political stance against academic colonization (Aarseth, “Computer Game Studies”; Eskelinen, “The Gaming Situation”). In the end, these discussions have ensured that the unique properties of videogames become a foundational topic in game studies.

Not all games can be accounted for in terms of narrative—understanding games such as *Tetris* as a narrative is a prototypical example of how games and stories do not necessarily go together (Eskelinen, “Towards Computer Game Studies” 37). *Tetris* was not originally designed to create a narrative experience, neither does the recounting of a play session of *Tetris* make for a particularly interesting one.

A recurring argument against considering narrative in games is that narratives can be interchangeable, therefore games do not need a story in order to help the player understand the game (see Myers). Some games have a fictional setting superimposed on a set of mechanics. For example, the game *Breakout* for Atari 2600 presents itself as a game where the player has to break brick walls with a ball and a paddle (*Breakout Game Manual*), whereas in *Super Breakout* the paddle is supposed to be a space ship, the bricks are a force field, and the ball is the blast that will break through the force field.<sup>7</sup> The narrative premise of the game, although completely different, does not do much to help the player understand what she has to do in the game.

---

<sup>7</sup> This example is taken from Salen and Zimmerman (*Rules of Play* 377).

On the other hand, there are games that integrate stories, where progressing in the gameplay is linked to advancing in the story. Adventure games are the prototypical example of how the story can drive gameplay, and the antithesis of the examples listed above. For example, King's Quest V resorts to fairy tale conventions to set up its premise; its gameplay includes challenges such as defeating a witch, surviving a desert, breaking a princess' spell and melting the heart of an Ice Queen. Rather than focus on how videogames may be incompatible with stories, this dissertation will deal with their points in common.

The commonalities between games and stories are not obvious, however. Games are systems that require the player's input, so that the player has to make decisions along the way. In stories, the decisions have already been made by the author; the audience has no choice or influence on the events of the story. Even in the case of stories offering the audience a choice, such as the *Choose Your Own Adventure* book series, the choices can only be made at specific points in the story, and they are a given set that is also authored.

Choice seems to be the problematic difference between games and stories. The issue is *who* makes the choices: the game designer or the player. The more story-driven a game is, the more restricted player choice becomes. Adventure games are not widely acknowledged for providing the player with choices, even in the case of having multiple ways to be traversed. The "gameness" of adventure games does not reside in player choice, but in puzzle-solving, which involves finding a specific solution to a problem. There is usually only one way to achieve that solution, giving the player little room for choice or providing creative solutions. The single solution facilitates the integration of the story, but also constrains the potential interaction of the player.

Stories thus work as a constraint, and games usually thrive on constraints. The player has to fulfill a certain set of conditions established by the rules, and those conditions must also be part of the story of the game. Stories are therefore a game design challenge, so that interacting with the game system should produce a satisfying story. So far, this has been achieved by curbing the agency of the player.

It is unsure whether games can strike a balance between providing the player with agency and generating an engaging story; a good start will be studying what adventure games have already done. Once we understand this relationship, we can suggest better avenues for innovation. Maybe the result is neither a new story form nor a game genre, but a new type of digital media artifact altogether.

Videogames are already challenging traditional concepts of both games and narrative. The technological aspect has created new ways of social play (e.g. MMORPGs), innovative variations on pre-existing games (e.g. *Tetris* uses the Pentominoes pieces, generating new gameplay by adding movement and time constraints), and creating worlds in which the player can play. Videogames are not narratives in the traditional sense, since neither the player nor the game makers have total control over the events in the story. The events result from the player's interaction with the system of the game. The system is authored by the game makers; they may guide the player to perform certain actions and providing specific choices, but they do not have direct control over what the player may decide to do with the system. This mediated relationship between the author and the audience of videogames is different from non-digital media, since the audience takes part in the unfolding of the events of the game. There are certainly many points of connection between narratives and games; by

exploring how traditional models of narrative are subverted, we are exploring a new media form that might be in between games and narrative.

### **3.3.Narrative Experience**

A recurrent argument in favor of considering videogames a new kind of narrative is that the gameplay experience can be recounted as a story. Bringing together theories from linguistics and cognitive science, Branigan affirms that “narrative is a perceptual activity that organizes data into a special pattern which represents and explains experience” (Branigan 3). Narrative schemas constitute a way of understanding our experience, so it seems natural to apply them to a gameplay experience; they can also explain some of the benefits of incorporating stories in games.

This narrativization of experience is a proven concept, but its contribution to understanding games can also be abused. It is all-encompassing, so its unqualified application can lose the nuance needed to account for games in narrative terms. Narrativizing a play session of *Bejewelled* is problematic—by equating videogames with all other experiences, we can lose sight of how they may be innovative and different from other media.

If we want to understand videogames in narrative terms, we must also acknowledge the *intentionality* on the part of the producers of the game to make a game that can be understood as a narrative. Playing a videogame is a designed experience, since it is the result of interacting with an authored system; thus the story becomes part of that experience. It is not enough that players narrativize their experience: the game makers should either create a system where the events will generate a story, or embed a story in the game, or a combination of the two. Game design can help form and reinforce

narrative schemas through the design of the game. What is relevant to this dissertation is how videogame design, the process of making a game, can harness narrative as part of the gameplay experience.

For our purposes, game design is the process of devising the system of a game, establishing what the player can or cannot do. The player interacting with the system creates an aesthetic experience (Hunicke, LeBlanc and Zubek; LeBlanc); therefore game design is also experience design. In regard to videogames, the system is not the rules of the game alone, but also how it the system implements the fictional world as a simulation (this will be covered in more detail in Chapter 5). There resides the potential to reconcile games and stories.

Branigan's approach can only account partly for the coalition of games and stories. Narratology, the study of "the nature, form and functioning of narrative (regardless of medium of representation)" (Prince 66), is a good complement to understand the relationship between games and stories.

### **3.4.Narrative: Story vs Discourse**

The polysemic term "narrative" is the first problem in this discussion. Gerard Genette lists three of the possible concepts that the term can refer to (25-26):

- the oral or written discourse that undertakes to tell of an event or series of events.
- the succession of events, real or fictitious, that are the subjects of the discourse, and [...] their several relations of linking, opposition, repetition, etc. "Analysis of narrative" in this sense means the study of a totality of actions and situations taken in themselves, without regard to the medium, linguistic or other [...].
- the event that consists of someone recounting something: the act of narrating taken in itself.

Genette's definition illustrates the problems of invoking the term "narrative" even in the context of literary works, and accounts for the different "aspects of narrative reality" (27). The first definition corresponds to *narrative*, "the signifier, statement, discourse or narrative text itself" (27); the second is the *story*, "the signified or narrative content" (27), and the third definition refers to the act of *narrating*, "the producing narrative action, and, by extension, the whole of the real or fictional situation in which that action takes place" (27).

"Narrating" is a problematic concept if we apply it to videogames, since it assumes that there is narrator telling the story. Buckles argued that there is a narrator in *Adventure*, since there is a voice informing the player of what happens in the depths of Colossal cave ("The Computer Storygame Adventure" 140-151; "Interactive Fiction as Narrative"). Given their text-generating nature, it is easier to draw parallels between literary texts and text adventures. However, the act of narrating does not seem to quite account for how *Myst* would be narrative, even when it is a story-driven game. Other games, from *Pac-Man* to *Tetris*, are difficult to conceive as acts of narration as well. Given their participatory nature, however, it is easier to think of them in terms of performance activities. Not all performance activities have to include a narrative (e.g. dancing a tango), but some of them do (e.g. Shakespearean theatre); in a similar way, videogames may or may not have a narrative. In short, videogames do not generate an act of narrating, but of performance.

*Narrative*, meaning the discourse that represents the events, is therefore a result of the performance rather than telling. This goes back to Aristotle's basic dichotomy between diegesis (a narrator telling the events) and mimesis (the events are enacted).

Ryan picks up on this division by stating that the discourse can take the shape of different semiotic manifestations, as *diegetic narration* or *dramatic narration* (“Beyond Myth and Metaphor: The Case of Narrative in Digital Media”). Dramatic narration has been invoked as the mode in which narrative discourse takes shape in games (Murray, “From Game-Story to Cyberdrama”; Mateas). However, how dramatic narration relates to the performing in a videogame has not been explored in the depth it deserves. The nature of that performance and its relation to narrative will be explored in more detail in the following chapter.

### 3.5. Defining Story

Narratology has made a distinction between story and narrative through a series of dichotomies: “The *content* plane of narrative, as opposed to its *expression* plane or discourse; the “what” of narrative as opposed to its “how”; [...] the *existents* and *events* represented in a narrative” (Prince 93). This section deals with the content, the “what,” the existents and events. The narrative as expression / discourse, the “how” of videogames is performative, and will be dealt with in the following chapter.

Genette’s second definition refers to the *story*: “the succession of events, real or fictitious, that are the subjects of the discourse, and [...] their several relations of linking, opposition, repetition, etc” (25). The story is the signified that is mediated and presented into a discourse. Videogames present a series of events—some are a result of the rules of the game in action, others may be pre-set by the makers of the game. This definition also makes explicit that the events are independent of the medium, which allows us to apply it to videogames. Finally, the events are related (“linking, opposition, repetition”), which provides an underlying and observable structure.

The intentionality on the part of the game makers to include a story is expressed in the design of the game. The story as content comes from the player interacting with the designed system, or from the embedded story of the game. The embedded story can be the prescribed set of steps that the player has to complete, and constitute the events of the game. Cut-scenes, text descriptions, comics, and other forms of traditional storytelling can also be embedded in the game. Both types of embedded stories are the predominant type found in adventure games.

### **3.5.1. Causality**

There is another aspect of story that is relevant to videogames—the causality between events. Genette already included the relationships between events of the story, but the nature of those relationships is abstract (“linking, opposition, repetition”) (25). Forster makes a distinction between *story*, “a narrative of events arranged in their time sequence,” (27) and *plot*, “also a narrative of events, the emphasis falling on causality” (86). This dichotomy, although at odds with other narratological definitions given above, emphasizes the importance of causal relations between events as a way to connect them, rather than their chronological order. This dissertation does not use the term *plot* in Forster’s sense, but considers *causality* as an important factor in the connection between stories and games. When the player interacts with the game, her actions have consequences, which help the player understand both how to play the game and how the world works. Causality is also vital in adventure games, particularly in the concatenation of puzzles—one solution will cause other puzzles to appear, or provide the means to solve other puzzles. Causal connections help create an experience that is coherent to the player.



### 3.5.2. Existents and Events

Chatman identified the “existents and events” as the two elements that make up the content of the story. These elements contain other components, as illustrated in Figure 1 (Chatman 19).



Figure 1: Diagram of the different components of the narrative text.

Events are “changes of state”; if a character causes the change, it is considered an *action*, whereas if the change happens to the character, it is considered a *happening* (44-5). The existents of the story are the *characters* and *setting* of the narrative (Chatman, 19).

Existents seem to be an obvious common feature of games and stories: the game takes place in a specific environment, which provides the setting. Characters are not a compulsory element of videogames (as our usual outlier *Tetris* may prove), and their

absence usually marks that the game does not have narrative intent. Videogame characters are game entities, agents that carry out actions according to a designed set of behaviors and respond to player's input; most importantly, they give the appearance of life. Game entities are different from game objects, which are props or tokens in the game. Objects also have behaviors attached to them, but they do not seem to have a will of their own.

Chatman does not seem to take into account the relevance of objects in the story. From Desdemona's handkerchief to *Citizen Kane's* Rosebud, objects have values attached to them that can be key in the unfolding of the events. As we will see, objects and their properties are a defining element of a performance activity; in adventure games, object manipulation is basic for puzzle-solving. For our purposes, objects will be considered part of the simulation of the game, with which the player interacts.

Space and time are the dimensions in which the story takes place. Since these dimensions are also performative in the case of games, they follow the rules established by the performance. In the following chapter we will expand on how space and time are identifying qualities of performance, and how videogames regulate them.

The existents of the story (characters and setting) and the events seem the obvious bridge between games and stories. The events of the story can be parallel to changes in the game state, whereas the characters can be the entities in the game who participate in those changes. The existents and their potential to bring about events are also part of a larger construct: the fictional world in which the story takes place.

### 3.6. The Fictional World

The concept of “world” is a promising avenue, since it encompasses both the existents of the story as well as the performative space where the game takes place.

“World” subsumes categories such as space, story, and game. Defining world broadly is not useful for constructing a function which tells us whether something is a world or not. It does, however, provide a place to stand for leveraging tools from across disciplines, and thinking about the spatial, procedural, representational, and participatory qualities of digital media artifacts as worlds. (Gingold 2).

The multiple aspects of the concept “world” need to be qualified. For starters, the worlds of videogames are fictional worlds. Fictional worlds are “imagined worlds” (Juul, *Half-Real* 122)<sup>8</sup> where the actions and events of the game take place; “fictional” thus refers to something fake pretending to be real (Klastrup). The fictional world provides a frame of reference to the player’s actions; the world becomes the context for the events and actions of the story, as well as of its existents.<sup>9</sup> Thus the concept of fictional world is essential to understand the relationship between story and game.

A canonical example of a fictional world is the *Star Wars* universe—its world was originally created for a movie, but it has since been incarnated in different media, from novels to toys to videogames. The fictional world can define a specific time and place (“A long time ago, in a galaxy far, far away...”), but it is much more than a

---

<sup>8</sup> Juul usually refers to the fictional world as “fiction,” which can be misleading. As Aarseth remarks that “fiction” used in this sense is an ill-defined term (“Doors and Perception”). Aarseth seems to equate fiction with non-interactive content, with a pre-determined story that the player cannot affect. On the other hand, “fiction” is another term used to refer to story (Prince 30). For the purposes of this study, the term “fiction” will be avoided to prevent confusion; “fictional world” and “story” are the more precise terms that will be used throughout.

<sup>9</sup> Fictional worlds are the cornerstone of transmedia storytelling (Jenkins, *Convergence Culture*), where the different stories of a world can be incarnated in different media: movies, films, comics, TV shows, toys, or games, for example.

setting—fictional worlds have to be inhabited (Klastrup; Ryan, *Narrative as Virtual Reality*). Characters complete the basic components of a story, since they are participants in the events of the story. They also represent the link of the audience/reader/player with the fictional world (more on this in Chapters 8 and 9).

Story-driven games have better-defined fictional worlds than games with no story. It can be argued that games such as *Tetris* or *Bejewelled* do not have a fictional world; they look more like a board game and, more importantly, they do not have characters who are active agents in the actions in the game. On the other hand, a game may have a fictional world, but it may not be story-driven. For example, a racing game such as *Wipeout* presents a futuristic world, with technology that allows cars to fly and reach extremely high speeds. However, the game does not have an embedded story (e.g. no cutscenes, the set of steps to advance in the game is to keep winning races), and how the player manages to win the race, although dramatic, are not actions that bring about a change of state other than winning or losing.

A key feature of fictional worlds is that they are incomplete, as Juul notes (*Half-Real* 122). It is impossible to figure out all the details of a world; it will always be smaller than the real world. Ryan refers to theories of Possible Worlds, which establish that fictional worlds will use the real world as reference, but their size and complexity will always be smaller than ours (*Narrative as Virtual Reality* 99-105). Even in the case of worlds whose story and documentation are as exhaustive as J.R.R. Tolkien's Middle Earth, there will always be aspects of the world that will not be covered. For example, it is not clear what the currency of Middle Earth is, and one can only glean how the economy works; we know there must be one because we use the real world as reference.

This information is crucial for an MMO such as *Lord of the Rings Online*, so the game features the types of currency and bartering that can take place. The gaps in the fictional world can be filled out by the player, and also offer the possibility to expand the world. It is important to keep in mind this incompleteness—in the same way an invented world cannot create every single detail, its implementation as a simulation will have to be incomplete as well, focusing only on the aspects that are relevant to the interaction, and simplifying or omitting others (more on this in Chapter 5).

At times, the player does not need to fill those gaps in order to understand the whole fictional world; other times, the player fills out the gaps by comparing it to similar ones. For example, in *Super Mario 64*, at the beginning of the game Princess Peach invites Mario to have cake at her castle; when the game starts, Mario learns that the Princess has been kidnapped by Bowser (yet again). In order to rescue her, Mario has to pick up power stars. It is difficult to understand the fictional world from what we see in the game. What do the power stars have to do with rescuing Peach? Why did Bowser kidnap Peach? Why does Mario have to rescue her alone, while all her mushroom-like subjects stand around hidden in the castle? If Peach is the princess, who are the queen and the king? The fictional world of *Super Mario 64* invokes folktale conventions in order to provide motivations for the character—the kidnapped princess that the hero has to rescue from the villain is a prototypical plot structure taken from Vladimir Propp. The familiarity of the story structure disguises the gaping holes in the fiction of the game, since folktales can also display similar omissions. *Super Mario 64* is not a story-driven game, which also makes the disconnect between the fictional world and what the player does less glaring.

### 3.6.1. Fictional World as Diegesis

The concept of the fictional world allows us to introduce the term “diegesis” as it is used in narratology. Rather than identifying discourse as “telling” in the Aristotelian sense, the diegesis is the fictional world where the main events of the story take place (Prince 20). From now on, the term *diegetic* will be used in the narratological sense.

The diegesis is the primary level of the narrative; what lies outside of it is considered extradiegetic. In the case of games, the diegesis is the aspect of the fictional world that the player has an effect through interaction; the main level at which the player performs. Game menus are extradiegetic, and so are the actions to save or load a game state (what Montfort calls *directives* in the context of text adventures (26)). The diegesis can also have different levels within it—a hypodiegesis is a narrative within the narrative, which may or may not share the same fictional world. In terms of games, this can be a story within the story of the game: in *Broken Sword: Shadow of the Templars*, the history of the Knights of the Templars is revealed through a series of manuscripts and explanations from experts—these stories are hypodiegetic. There can also be games at the hypodiegetic level, presenting a game within the game. For instance, *Sam & Max Hit the Road* include several mini-games, one of which is a Whack-a-Mole machine. The player has to win the game in order to obtain a necessary item as a prize; after winning, the player can go back and play the game whenever she wants, although there will be no more prizes.

Hypodiegesis will be covered more extensively in later chapters. The different narrative levels as integrated in the space will be tackled in Chapter 7; whereas the games within the game will be dealt with in Chapter 6.

### 3.6.1.1. Types of Stories in Relation to the Fictional World

The fictional world is the foundation for the story of the game. However, the nature of that story, and how it may or may not include the player's interaction, has been usually misunderstood. "Backstory" is frequently invoked in games to refer to the events that have happened in the world before the game starts. In most cases, the term is usually used dismissively (Myers), emphasizing its irrelevance to the game itself. The story in this sense is supposed to be a backdrop that can be removed without affecting the game, as we saw in the case of *Super Breakout*. However, there are games where the story cannot be removed from gameplay, as is the case with adventure games. Even when games and stories are integrated, the concept of backstory seems insufficient to account for the function of stories in games.

There are two different types of story depending on the involvement of the player with the fictional world: the pre-history of the fictional world, and the player's history.

Pre-history is the history of the fictional world, the events that happened in the world before gameplay starts, as well as the processes that have shaped the world and how it works, such as its economy or its social strata. Although similar to the concept of backstory, it encompasses past events, as well as how the world works, e.g. its economy, social classes, or religions. If magic is effective in the world, its workings are also part of the pre-history of the game (e.g. Voodoo magic in *The Secret of Monkey Island*, vs. fairy-tale magic in *King's Quest V*). The regulation of the fictional world is determined by past events, even if those events are never explicit in the game.

Player's history is what happens as the player interacts with the game; it is the story of the player's performance. These are the events during gameplay itself, as well as

the events triggered at specific points of the game. Thus pre-scripted events and represented events, such as cut-scenes or textual descriptions, also become part of the story as the player interacts with the game, because they are also part of the experience.

A cutscene does not cut off gameplay. It is an integral part of the configurative experience. Even if the player is denied any active input, this does not mean that the ergodic experience and effort is paused. (Klevjer, “In Defense of Cutscenes” 195)

Klevjer argues that the cut-scenes are part of our understanding of the fictional world—they set up the player’s expectations about what is going to happen (196-7). The cut-scenes can also be a reward for doing certain actions right, a spectacle for the player to enjoy. For our purposes, pre-scripted events (“micronarratives” is the term used in Chapter 7) such as comics, textual narrations or animations, are all part of the experience of playing the game. Pre-scripted events cater to the player as spectator, another aspect of gameplay as performance that will be expanded in the following chapter.

For the purposes of studying adventure games, represented events are an integral part of gameplay. From the descriptions of each location in text adventures to elaborate cut-scenes, these represented events are the responses to interaction which contextualize and give meaning to the player’s actions within the fictional world.

The basic distinction between pre-history and player’s history is necessary to identify what aspects of the story are relevant to the player during gameplay. While the pre-history is part of understanding the fictional world, and may not be made explicit in the game, the player’s history is the result of the player’s interaction, since she is setting off the events.



### 3.7.Conclusion

In spite of the differing nature of stories and games, we have laid the foundations of the relationship between games and stories. Narrative schemas can help structure the experience of the player, particularly when the game already incorporates stories. The basic distinction between story and discourse allows us understand how narrative in games is different from other media. While *story* refers to the content, the events that take place, *discourse* is how those events are represented. In games, the discourse is not a representation, as it is in written fiction or film. Although games can incorporate representation, in the form of cut-scenes, images, or textual descriptions, these representations are integrated in a larger discourse, which is the performance. The following chapter will deal with performative discourse of videogames.

Most importantly, the definition of story helps us identify the basic connection between games and stories: the events, the characters involved in them, and the setting where the events take place. The fictional world that hosts the events and exists thus provides a useful construct to bridge stories and games. The incompleteness of the world, as well as the types of stories that the world can host, will be factors in understanding how the world is constructed in the game, and how the player can narrativize her experience while performance.

The basic elements linking story and game are the characters, time, and place. These elements are incarnated in the fictional world of the game, a construct that sets the stage of events. The fictional world is part of the simulation of the game, which we will deal with in Chapter 5.

## **CHAPTER 4**

### **PERFORMANCE IN VIDEOGAMES**

#### **4.1.Introduction: Dramatic Discourse**

The previous chapter dealt with story as “content” – the “what,” the “existents and events” – but not with the expression, the “how”: the discourse, the expression of that content. As we briefly discussed, it is difficult to understand videogames as a series of events that is recounted. Following the Aristotelian division between diegesis and mimesis, Ryan points out that the discourse is not always necessarily recounted (diegetic narration), but can also be dramatic (“Beyond Myth and Metaphor: The Case of Narrative Media”). In a revised version of her 2001 article, Ryan expands on how narrative can be diegetic or mimetic / dramatic.<sup>10</sup>

The diegetic conception presupposes the existence of a narrator or storyteller addressing an audience. In the mimetic or dramatic conception of narrative, by contrast, there is no need for a storyteller. Narrative comes into existence not by being told but by being enacted. [...] In the mimetic model, the computer is not a storyteller but a character who interacts with the user in such a way that the user will regard their interaction as a story. (“Beyond Myth and Metaphor: Narrative in Digital Media”)

Ryan here contrasts two different models: one is storytelling, and the other is enactment of events. These different approaches will also have different consequences in the design of digital media—in the first, the interactor is the audience, who cannot influence the story, whereas in the mimetic model, the interactor is a participant and it is

---

<sup>10</sup> Ryan is framing this discussion in terms of interface metaphors, as different approaches to understand the participation of the interactor with digital media.

the interaction that generates the story. However it seems more appropriate to understand games as mimetic narration, since they are based on actions and their enactment, both by the computer and the player. Thus videogames may be considered a new form of dramatic discourse, as a series of enacted events.

Text adventures present a tricky case, since the events themselves are recounted verbally. Buckles discusses the role of the narrator in *Adventure* at length (“The Computer Storygame *Adventure*,” 140-58); in that discussion, she finds the role of the narrator as ambiguous, both as an “author narrator or a character narrator (some type of electronic guide)” (156). Buckles concludes that “the story does ‘tell itself’”, where the narrative voice describes what happens, while the emotional reactions, rather than being described in the text, are left to the reader / player (156). It is possible to account for text adventures / Interactive Fiction as diegetic narration, given their links with literature (Buckles, “The Computer Storygame *Adventure*”; Montfort, *Twisty Little Passages*). However, storytelling can also be a performance act—think of oral storytelling. In a similar way, we can also understand text adventure games as performance. As we will see below, the computer as narrator is also a performer, since it produces the text procedurally (according to the rules of the performance), and in reaction to the input of the player.

Dramatic models have been repeatedly invoked as a model to study virtual environments, as well as to suggest new dramatic forms based on digital media. Brenda Laurel was the pioneer in applying Aristotle’s *Poetics* to account for the processes in Human-Computer Interaction. Gonzalo Frasca’s *Videogames of the Oppressed* proposed adapting basic strategies from drama theorist Augusto Boal into videogame design. Janet

Murray referred to the potential of videogames as cyberdrama (*Hamlet on the Holodeck*; “From Game-Story to Cyberdrama”), a new narrative form that consists of an immersive story in which the player takes part. However, in spite of invoking the term, Murray and those inspired by her theories (see Mateas and Stern) appeal to very specific dramatic theory beyond Aristotle or Augusto Boal.

In terms of game design, *drama* is a repeatedly invoked term (Fullerton, Swain and Hoffman; Salen and Zimmerman; Spector), usually referring to the different strategies to create uncertainty and tension in gameplay. LeBlanc specifically discusses the concept in terms of “dramatic arc”, where gameplay must contribute to building tension that is brought to a climax and then a resolution (LeBlanc).<sup>11</sup> “Drama” implies an emotional reaction, as well as a staged/designed one—the events are only relevant as far as they contribute to create an experience. In game design terms, “drama” usually refers to a conflict, and the tension created around that conflict, usually building on the uncertainty of the outcome.

A dramatic model seems therefore germane to the study of events and existents in videogames, both from a narrative and a game design standpoint. Rather than referring to dramatic theories for videogames, this dissertation resorts to performance studies to create a framework, which are then applied to adventure games. Performance studies is an inclusive field, which does not limit itself to one medium, and intersects with multiple disciplines. Thus, it seems almost natural to include videogames as one of the objects of study, especially since game studies is also an interdisciplinary field.

---

<sup>11</sup> It is significant that LeBlanc always talks about “drama” in videogames, rather than “narrative” or “storytelling.”

The origins of adventure games are performative too. Both *Adventure* and *Zork* have their roots in table-top role-playing games, which is a performance activity given its links to both games and theatre (see Mackay). This chapter outlines what it means to study videogames as performance, what that implies for the different types of stories in games, and how this is exemplified in adventure games.

#### **4.2.What is Performance?**

Performance studies devotes itself to how human action takes place and in what context. In semiotic terms, the field deals with performance as a process of making meaning on the part of the originators of the activity and their audience.

“Performance” can have different definitions in English: from carrying out an action, to the degree of efficiency with which something is done, to more specifically an entertainment event (a theatrical play, a music concert, a dance) that is presented in front of an audience.<sup>12</sup> Only the first and second definitions are relevant here. Performance is the study of actions, of *how* things are done. Theatre, music and dance activities require a certain skill to be performed, but the skill is not what is relevant: it is the fact that those activities take place in front of an audience. Without an audience there is no performance. Schechner defines performance as “showing doing”: pointing to the activity itself (*Performance Studies: An Introduction*, 28).

Schechner identifies five different types of activities that can be dealt with as performance: play, games, sports, theatre and ritual (Schechner, 7). The main

---

<sup>12</sup> “Performance” is a tricky term to translate into other languages. It does not have a direct equivalent in Spanish, my native language. All these three concepts are translated into different terms, and none has a related cognate.

characteristic of all these activities is *that they take place separate from everyday life*. The values and rules of everyday life do not apply during the performance, because the activity takes place within much quoted “magic circle” that Huizinga referred to.

It is certainly convenient for our purposes that performance studies already deals with activities that relate to videogames: play and games on the one hand, and theatre on the other. This pre-existing relationship also allows us to establish points of comparison between games and narrative dramatic forms (as opposed to theatrical activities where there is not a story, e.g. contemporary dance, Balinese theatre).

There are certain aspects of performance that will not be contemplated here, because although related, they are not directly relevant to the framework proposed. One of those aspects is social performance and the shift between different social frames, as studied by Erving Goffman (*Presentation of Self, Frame Analysis*). The dissertation does not deal in depth with the ontology of the “magic circle” to question its existence, nor how performance may have an effect on everyday life (Bogost; McGonigal). What follows is a foundation that will allow us to deal with those issues in a different context, beyond the scope of this dissertation.

#### **4.2.1. The Basic Qualities of Performance**

Play, games, sports, theatre and rituals have five basic qualities in common: a special ordering of time, a special value attached to objects, non-productivity in terms of goods, rules, and performance spaces (Schechner, *Performance Theory* 8-19). These qualities allow us to identify videogames as another type of performance activity, as we will see below.

#### 4.2.1.1. Time

Time is adapted to the performance, and thus liable to be structured in different ways (Schechner, *Performance Theory* 10). Time can be determined by the completion of a series of steps, no matter how long it takes (e.g. a jigsaw, a religious ceremony, a baseball game)—Schechner calls this *event time*. There can also be a specific amount of time within which the activity takes place, so its length is predetermined. This *set time* is what we find in basketball, or a rite that must take place on a specific day of the year. Finally, *symbolic time* takes place when the activity represents another period of time, either longer or shorter. Theatre presents a clear example of symbolic time: a play may represent the events that take place over several days in a couple of hours.

All these types of time apply to videogames; similar terms have already been applied in game studies (Juul, *Half-Real*; Zagal and Mateas). Time in games can be defined by how long it takes for the player to achieve its goals. Event time is particularly common when the gameplay is mainly puzzle-solving, as is the case of adventure games like *Myst*. A game can establish a time limit to reach the goal, as in *Marble Madness*, where the player has to bring a marble from the start to the goal within a very tight time limit. The time of the simulation can also represent a different length of time from that of the real world, usually shorter. Examples of these are *Grand Theft Auto: Vice City*, where a day lasts 20 minutes, or the day/night cycles of MMOs such as *World of Warcraft*. Chapter 10 deals with time and its regulation in more extensively, defining a temporal framework based on previous works on the topic by Juul, and Zagal and Mateas, and applying it to adventure games.

#### 4.2.1.2. Objects

Objects in everyday life have some sort of worth; borrowing Marxist terms, they have use value (they are useful, e.g. tools or reference books, basic furniture) or exchange value (they are worth money or a prize, e.g. an antique object) (Sturken and Cartwright 199). In performance, objects acquire a meaning and value during the performance that does not correspond with its value in the world outside of it: “[...][T]hese objects are of extreme importance, often the focus of the whole activity.” (Schechner, *Performance Theory* 11). Think of a sword in a theatrical play, which may look elaborate but it may be made of painted wood—its exchange value and use value in everyday life is very low, but it has a high use value in the play, since it can kill other people in it. The money in the board game *Monopoly* is an even clearer example, since its exchange value outside of the game is nil, but it has a very high exchange value in the game. We can continue extending the comparison to game tokens, or the ball in certain sports (e.g. football, basketball)—as Schechner notes, the disparity between the value within the performance is another indicator of the separation between performance and every life.

In videogames, the value of an object is related to how much of an effect it has in the game. For example, in *The Legend of Zelda: Phantom Hourglass*, there are many items that have a high use value. Heart containers and potions improve the health of the player character. The grappling hook allows Link to reach places that are otherwise inaccessible by making a tightrope between two poles, or by dragging the player character to a heavy object when it reels back. The player can also find rupees throughout the quest, which have exchange value in shops and allow the player to acquire new objects, which in turn may have a higher use value (e.g. potions that restore more life). In



the case of adventure games, objects are essential, since they are usually part of puzzles. Object-hoarding, i.e. picking up items and putting them in the game's inventory, is a common player strategy because using objects allows the player to advance in the game. Finding an object and combining them with another, or giving an object to a character are two basic mechanics of adventure games.

#### 4.2.1.3. Non-productivity

The performance does not produce money or other goods that may be useful outside of it. This aspect of performance, based on the definitions of “play” by Huizinga and Callois, is debatable (Schechner, *Performance Theory* 11-12; Juul, *Half-Real*, 35-6). Gambling and professional sports are two examples that challenge the concept of non-productivity, since playing them produces a material benefit (or loss) in everyday life. Non-productivity is a quality of performance that is easily challenged, particularly by games and rituals.

Many videogames take place within a virtual environment, which makes it difficult for its outcome to be anything material or in the real world.<sup>13</sup> On the other hand, there are also professional players that make a living in videogame competitions. MMORPGs also challenge the division between the virtual world and everyday life: there is a market of *World of Warcraft* high-level characters for sale in the real world.<sup>14</sup>

Adventure games are non-productive, closer to a theatre play than to professional sports. As a genre, they do not lend themselves to having players compete against each

---

<sup>13</sup> Note that we are only referring to material goods; other effects may transcend the videogame space, as in learning or improving eye-hand coordination.

<sup>14</sup> See <http://www.playerauctions.com>

other, also because they are single-player games. Players may solve a puzzle faster or slower, but since there is usually only a few ways in which it can be solved, it makes no sense to have professional puzzle game players or betting around gameplay.

#### 4.2.1.4. Rules

Performance activities are regulated; Schechner notes that rules specify how the activity takes place and, more importantly, set the activity aside from everyday life: “[T]he rules are designed not only to tell the players how to play, but to defend the activity against encroachment from the outside.” (Schechner, *Performance Theory* 13). The goals of the activity are also implicit in the rules: from what the winning conditions of the game are, to making the gods hear one’s pleas. In theatre, the rules are part of the dramatic text, or the set of movements that will constitute the theatrical performance, as we will see below.

Rules also establish how a certain activity *should* be (Schechner, *Performance Theory* 12-13), and although they can be adapted for special circumstances, they define what the ideal form of the activity should be. For example, in a school playground, the dimensions of a football field may be adapted to whatever room is available, and the distance between the goal posts may be established by the space between two waste bins.

Play presents an interesting challenge to the concept of rules—Schechner seems to understand play as “free activity”, where the participants regulate their own behavior, usually as they perform it. This divergence seems to relate to Callois’ dichotomy between *ludus* and *paidia*. The term *paidia* refers to free play, activities that are improvisational, spontaneous and unregulated, like a cat playing with a ball of wool. Callois alludes to the potential destructive (or at least disruptive) nature of *paidia*, which is lead by the impulse

to touch and manipulate objects at hand, and then set them aside; thus, experimentation seems to be an important part of the activity. *Ludus*, on the other hand, is play that is arbitrarily constrained in order to achieve a specific goal. This is regulated play, which requires the player to acquire specific skills in order to reach that goal. Puzzles, and games that have a set of rules (from Poker to *Monopoly*) belong to the *ludus* category.

When it comes videogames, rules are enforced by the code of the program(s) that make the game. Thus, the rules of the game are always the same, “how the game should be” is upheld by the computer itself. Variations have to be supported by the code, such as different levels of difficulty. Cheat codes and hacks can also change the nature of the game, but that makes the game different from its proper form.

Adventure games present a particular case, because the rules establish how the player can interact with the world (e.g. how to navigate, what input commands are acceptable). The goals, on the other hand, are set by the story, and consist of solving the puzzles in the game. For example, in *Deadline*, the goal is to find out who killed Marshall Robner within twelve hours (in-game time), which involves the player obtaining information by investigating the crime scene. The time limit in this case is a regulation similar to time periods in sports, but how the crime is solved depends on what clues the player gathers, and how she infers the solution from those clues.

Rules will be discussed in more detail in the following chapter. Their relation with the fictional world of the game is what establishes how the simulation of the game works. In the case of adventure games, figuring out what the rules are explains how the player can interact with the game.

#### 4.2.1.5. Performance Spaces

Performance activities are a shared and regulated experience in space. The rules are effective only within a specific space, which at times is purposely designed for that activity, thus reinforcing the idea of separation from the rest of the world. Religious temples, sports stadiums, theatres and game boards are all performance spaces. The idea that play is separated from real life is epitomized by the concept of the “magic circle” (Huizinga; Salen and Zimmerman) which is a fairly extended (and abused) concept in game studies: a safe place where the rules of the game are in effect.

However, the performance itself can create the space ad hoc, as was the case of playing football in the playground above. A game of tag turns a park into a playground, the players themselves setting the limits of how far they can run; street theatre can turn a corner into a stage. The “magic circle” is transportable and flexible, and as permeable as the performers make it to be.

In the case of videogames, the separation between real space and performance space may seem clear, since the performance space is represented on the screen and does not exist in the real world. However, videogame space must also extend beyond the screen, since the input of the player (e.g. keystrokes, joystick movements, or shaking the controller) takes place outside of the represented digital space. Nitsche suggests five planes to analyze videogame space: the rule-based space, defined by mathematical rules (e.g. physics, AI); the mediated space, how the space is represented visually<sup>15</sup>; fictional space, which is the space that the player reconstructs in her mind; play space, which is the physical space occupied the player and the game hardware; and social space, where

---

<sup>15</sup> Nitsche here seems to overlook that the space can also be presented aurally as well.

several players are interacting with each other (Nitsche, *Video Game Spaces* 15-16). This multiplicity of planes indicates the complexity of the performative space, where the transitions between planes must also be regulated.

This multiplicity of spatial planes in videogames also accounts for the duality of player as performer and spectator in her own game—remember, performance spaces include both the performer and the audience. The framework below addresses this duality, since it allows us to compare videogames with other performance activities.

As we will see in Chapter 7, the performance space of adventure games is one that encourages exploration. Part of the game may be reconstructing the configuration of that space—by drawing a map, for instance. The spatial design of adventure games shapes how the player performs within it. The limits of the space are usually restricted to the digital space itself, although some adventure games used physical props outside the game. That was the case of the “feelies” included in some Infocom games, which doubled as keepsakes and anti-copy protection. The “feelies” usually included information that was essential to complete key puzzles of the game. For example, the feelies of *The Lurking Horror* included a password to log in to a computer in the game; *Plundered Hearts* included a fake bank note, whose image allowed the player to find a secret passage (the image was obviously not described in the game).

Now that we have established how videogames are also a performance activity, we will outline the framework that allows us to account for the *process* of playing videogames as performance, and how it relates to the different types of stories in games.

### **4.3. Performance Framework for Videogames**

This framework intends to lay down the foundations to study videogames as

performance, so its relevance extends beyond this dissertation. The foundation of the framework is the theatrical performance model, which will be then applied to digital media and games. Since Schechner does not deal with games in much depth, we will make use of the terms set up by the MDA framework (Hunicke, LeBlanc and Zubek). MDA was conceived as a tool to bridge game design and development with game criticism and technical games research; by including these terms in our framework, we can extend that bridge to performance studies.

Theatre, interacting with software and playing games constitute the three basic layers of our performance framework. With this approach, we are not trying to equate theatre to games, but rather to help us understand videogames in terms of theatre, and to identify the differences introduced by this new media form. The goal of the framework is to plot the relationships between pre-existing concepts that define each of the layers as the basis of a performance activity.

There are several reasons to use theatre as the basic reference model to study videogames as performance. First of all, it is a familiar performance activity, so it is relatively easy to give accessible examples. Theatre also has a long, rich and complex history, which will allow us to find exceptions to the rule that may relate to videogames. As we discussed above, theatre has already been used as a reference model in game studies and related fields (Laurel; Frasca, *Videogames of the Oppressed*; Murray “From Story-game to Cyberdrama”, Mateas), so the relationship has already been established. Callois also includes theatre in his seminal work *Man, Play, Games*, as mimicry play. Significantly, he considers it a type of *ludus* activity, i.e. organized and regulated, just as videogames are.

	Play	Games	Sports	Theatre	Ritual
Special Ordering of Time	Usually	Yes	Yes	Yes	Yes
Special Value for Objects	Yes	Yes	Yes	Yes	Yes
Non-productive	Yes	Yes	Yes	Yes	Yes
Rules	Inner	Frame	Frame	Frame	Outer
Special Place	No	Often	Yes	Yes	Usually
Audience	Not necessarily	Not necessarily	Usually	Yes	Usually
Performed by Group	Not necessarily	Usually	Usually	Yes	Usually
Symbolic reality	Often	No	No	Yes	Often
Scripted	Sometimes	No	No	Yes	Usually

*Figure 2: Performance chart (excerpt).*

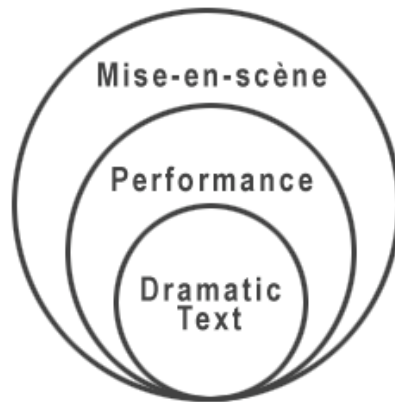
Theatre also has the most elements in common with other performance activities (Partlett 2). Schechner gives evidence of this through a chart (see Figure 2), where we can observe how the elements that are present in theatre also appear in at least two other types of performance activities (“Performance Theory: An Introduction 16). As we just saw, the special ordering of time, special value of objects, rules and non-productivity are common elements to all performance activities. The chart shows that not all performance activities required a special place built for them, although it is usual in most activities—in Chapter 7, we will explain in more detail how performance can transform the space into a “special place.” The chart shows some additional qualities of performance that also extend to videogames apart from those already described above (audience, performed by

a group, symbolic reality, scripted). Some of these other qualities raise challenging questions; for example, some videogames can be played by a group, although not necessarily, and some games may consist of following a specific script of actions (more on this below). Videogames may also have audiences—think of professional videogame tournaments. Whether videogames represent a symbolic reality is a complex question beyond the scope of this thesis.

#### **4.3.1. The Theatrical Performance Model**

The theatrical performance model follows a tripartite structure, which accounts for the different components that are involved in theatrical performance. This model is based on the ones proposed by Patrice Pavis and Richard Schechner (*Performance Theory*). Pavis is a semiotician concerned with the *mise-en-scène* of written plays, which is the core of the Western tradition. Schechner, on the other hand, intends to encompass all traditions of theatre, including dance and plays that are not based on a written text. This framework uses Pavis' proposed definitions, and refines them with parallel observations from Schechner, who proposes similar concepts but aims at extending the theatrical model to other performance activities. Pavis' terms are used in order to be consistent with the three-partite division, and also because his three basic concepts map well with Schechner's. There are three basic components in this model: the *dramatic text*, the *performance* and the *mise-en-scène*. These components are interdependent, forming a concentric structure, as shown in Figure 3.





*Figure 3: The Theatrical Performance Model*

#### 4.3.1.1. Text

The dramatic text is the script that is pre-set before the performance starts. In theatre plays, this means the written verbal text of the play and the stage directions—this is the core of Western theatre tradition (Pavis). Schechner refers to the written text as *drama*, which can also be “[a] score, scenario, instruction plan or map,” (*Performance Theory* 71). Schechner distinguishes it from *script*, which is “the basic code of events,” the steps that are followed in the performance, e.g. what has to be said or sung, or how the participants move. With this distinction, Schechner aims at including traditional theatre, whose pre-set actions and text may be transmitted orally, rather than in written form. In Aristotelian terms, the text is potential and needs to be actualized by the performance. What it can get to be will be determined by the rest of the components of performance.

#### 4.3.1.2. Performance

Performance in this context refers to the actors playing, their voices and movements on the stage; Pavis calls this process *concretizing* the text (27). Schechner uses the term *theater* to refer to this component, “what the performers actually do during production.” (*Performance Theory*, 70). A play may be written for the stage, but it does not become a play until it is enacted. As Pavis notes, enactment does not mean “completing” the text, or translating the text into movement—it is the interaction between both the dramatic text and its concretization. A text can mean something completely different depending on the actor’s delivery and movement, even if the words are the same. For example, in Shakespeare’s *The Taming of the Shrew*, the actress can perform Katherine’s final soliloquy literally, to signify the submission of the character to her husband; the actress can also imply with gestures that she does not mean it, making her rebellious until the end (see Henderson).

#### 4.3.1.3. Mise-en-scène

The last component is the *mise-en-scène*, which is the confrontation of the dramatic text and its performance (Pavis 26).<sup>16</sup> This is the process by which the audience makes sense of the dramatic text and the actions on the stage. Schechner refers to this wider circle as the *performance*, “the constellation of events [...] that take place in / among both performers and audience from the time the first spectator enters the field of performance [...] to the time the last spectator leaves.” (70)

---

<sup>16</sup> Pavis’ use of the term *mise-en-scène* is different from how it is normally used in theatre and film studies, where it refers to how a theatre or film is staged or presented to the camera.

The audience is thus indispensable to theatrical performance, since it completes the process of making meaning. This is relatively easy to understand: if there is no audience in the theatre, the actors enacting the play are rehearsing, but it is not a show. If the text is being read to oneself, the reader can imagine the characters and how they speak and move, but since there is no actual enactment of the actions, there is no performance either.

Modern Western tradition usually establishes the performance as separate from the audience, although this division (the usually misunderstood concept of the “fourth wall”) is often challenged and subverted. There are kinds of theatre which encourage audience participation, such as puppet theatre for kids, or Boals’s method for the Theatre of the Oppressed; Brechtian theatre also encouraged playwrights and players to acknowledge the audience and talk to them, just as the chorus of Greek plays talked directly to the audience.

#### **4.3.2. Performance in Digital Media**

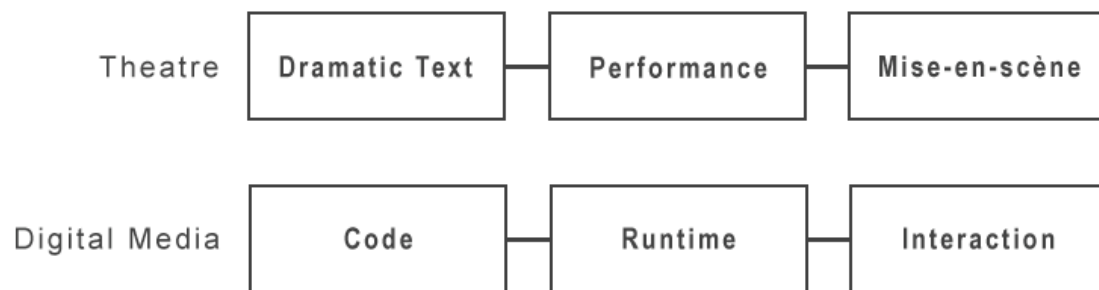
Videogames combine computer software with games; both aspects can be studied in terms of performance using the terms established by the theatrical model above. The computer becomes the performer, whereas the interactor is now the one to close the circle. The cybertext model proposed by Aarseth (*Cybertext*, Introduction) was also tripartite, and intends to incorporate the performative nature of cybertext.<sup>17</sup> Although it has some points in common with the model proposed here, the concept of cybertext

---

<sup>17</sup> For Aarseth, this structure is triangular, in his own words, it is a “feedback loop”. There is the *verbal sign* (the text itself in the literary sense), the *medium* where it is presented, and the *operator*, who is the person who interacts with the system (the interactor in our terms)—without the operator, there is no cybertext (as is the case in theatrical performance or games).

focuses on the production of the text itself, rather than the interaction with it, which is what we are focusing on.

The three components of computer software as performance are the *code*, *runtime* and *interaction*. These components are parallel to those in theatrical performance (see Figure 4 below).



*Figure 4: Parallels between Theatrical Performance and Digital Media*

#### 4.3.2.1. Code

The code of the videogame works like the text of a play: it is the pre-set data that the software is based on, and it is at the core of the computer software. It is the data on a disk or a hard-drive, the set of instructions that the computer has to follow. If the code contains errors, then the program will either not work properly or not run at all. As Klastrup mentions, the code is virtual, “is till something ‘in potentia’”, which does not become concrete and actualized until the code is used by someone (Klastrup 102).

#### 4.3.2.2. Runtime

Runtime refers to the computer executing the code, the program working as it is supposed to. It is the process taking place after the interactor starts the program (e.g. by typing a command, pressing the start button, loading the program to a browser). During runtime, the computer becomes a performer, as Ryan pointed out (“Beyond Myth and Metaphor: Narrative in Digital Media”). The computer is following the rules of the performance as set by the code. Unlike theatrical performance, where the actors can ad-lib or ignore the stage directions, computers must follow the code and cannot alter it.

#### 4.3.2.3. Interaction

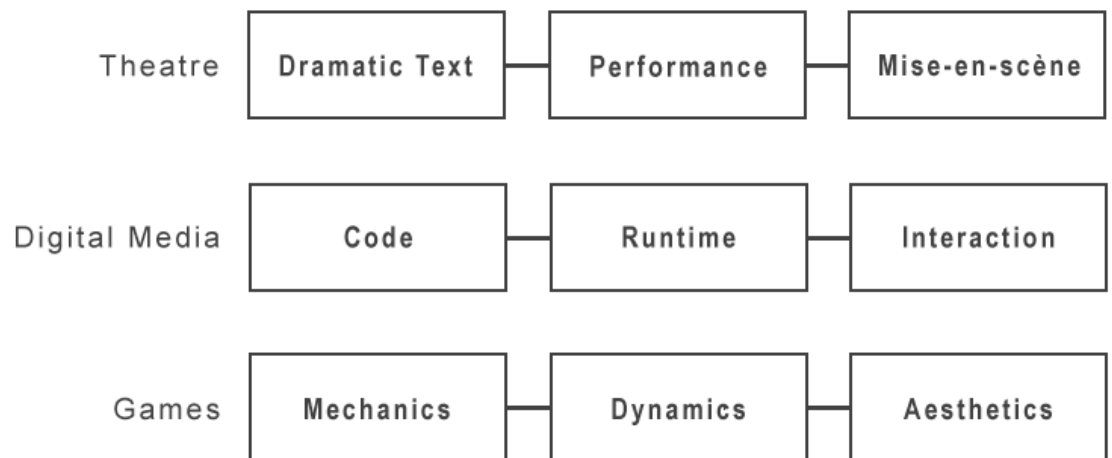
Similar to the *mise-en-scène* of a play, the interactor closes the circle of software as a performance activity. In the same way that theatrical performance does not happen without an audience, interactive applications (as is the case of videogames) may run but are not functional until there is input from the interactor. The difference with theatre is that the interactor does have an effect on runtime—the computer needs the input of the player to produce an output, following a conversation model (Laurel). The interactor is thus an active co-performer along with the computer. If in theatre the division between performance and *mise-en-scène* is often challenged and broken through, in software that breaking through is obligatory for the performance to take place.

### **4.3.3. Games as Performative Media**

Since Schechner does not specify how the theatrical model applies to games, we are going to use a pre-existing model, the MDA framework (Hunicke, LeBlanc and Zubek), to account for games as performance activities. MDA is intended to describe

games in general, not necessarily digital. This framework is particularly relevant because its understanding of games promotes a philosophy of game design, where each component relates to each other, establishes how game design can connect with player's experience. Therefore, MDA allows us to integrate concepts that are relevant to game design into our performance framework.

MDA stands for the three components of the framework: Mechanics, Dynamics, and Aesthetics. These three components are parallel to the ones previously established for theatre and digital media, completing the structure of the performance framework for videogames (see Figure 5).



*Figure 5: The Performance Framework for Videogames*

#### 4.3.3.1. Mechanics

Mechanics are the design counterpart of rules: “Mechanics are the various actions, behaviors and control mechanisms afforded to the player within a game context”

(Hunicke, LeBlanc and Zubek). That is, the mechanics provide the means to the player to perform in the game.

The mechanics establish how the player can perform in the space and with objects needed to play the game. Objects have special values inside the performance space; in the specific case of games they also have rules attached to them. Chess is a clear example of this: the King is the most valuable piece of the set—if it is captured or in a situation where it cannot escape capture, the player who owns it loses. Each chess piece moves in a specific way, determined by rules: bishops move diagonally, the King moves in eight directions but only one square.

Game mechanics are therefore parallel not only to the computer code, but also to the dramatic text—the pre-set text that constitutes the foundation of the performance.

#### 4.3.3.2. Dynamics

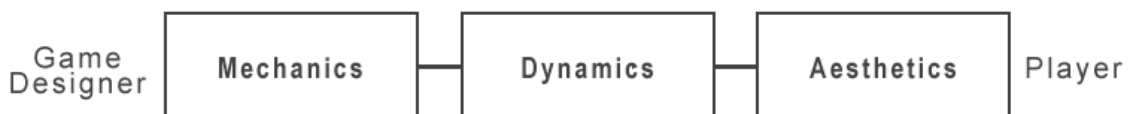
The dynamics of the text consist of how the mechanics play out, the events that take place when the game is played; they are the counterpart of the game system. Hunicke, LeBlanc and Zubek actually refer to it as run-time behavior, making the connection with the runtime of the software very clear. The dynamics of the game refer to how the rules are performed, and here is when the comparison with theatre is particularly relevant. The rules are not being enacted or translated, they are producing specific movements and behaviors with the objects of the game. As the player understands the dynamics, she comes up with different strategies. *Tetris* is a good example: players can leave a single column free, in order to accommodate a long piece and clear four lines at a time. However, the higher the blocks get while waiting for that long piece, the more difficult the game gets, since there is less and less room to maneuver.

The game dynamics are parallel to the code running – the system is set in motion both in the game and the software – and to the performance in theatre.

#### 4.3.3.3. Aesthetics

Aesthetics is the most ambiguous concept of the MDA framework. It is presented as the counterpart of “fun,” which in itself is a very broad concept. Hunicke, LeBlanc and Zubek define it as “the desirable emotional responses evoked in the player, when she interacts with the game system.” Aesthetics refers to the experience of the player while playing the game, which is the result of the interactions of the player with the system. Hunicke et al. refer to aesthetics as “kinds of fun,” listed as follows: sensation, fantasy, narrative, challenge, fellowship, discovery, expression, submission. A game can include several types of fun, as ways of engaging the player.

The relationship between the three components (mechanics, dynamics, aesthetics) is key to understanding games as performance. From the point of view of the designer, the mechanics generate dynamic system behavior, which creates certain aesthetic experiences (Hunicke, LeBlanc and Zubek). Figure 6 illustrates how the player stands at the other end of the cycle, starting with the experience created by the game, then devising the strategies to play the game, to finally figure out the mechanics of the game.



*Figure 6: The MDA Framework and Its Relationship between Designer and Player.*



Knowing what the rules of the game are does not mean that the player knows how the internal system of the game works. For example, in *Mario Kart: Double Dash*, the instruction manual lists the different items that appear randomly during the race. Only by playing the game can the player realize that the items are issued depending on where she is in the race, and which character she is controlling. If the player is at the back of the race, she will get power-ups that allow her to accelerate and catch up, if she is at the top, she will only get obstacles to put on the way of the racers behind her.

The MDA framework presents another parallel with theatre. The playwright can write the text, but eventually has little control over how the text is performed and how it is received. Unless the playwright actually works with the actors while they prepare the play, theatre companies appropriate the texts, cut them or extend them, ignore the stage directions, and eventually make it their own version. The audience can also make their own interpretation of it—a version of *A Midsummer Night's Dream* where the language has been updated for contemporary audiences may become very popular amongst younger audiences, but it may be considered blasphemous by theatre critics. In a similar way, the game designer also does not have direct control over the experience of the player either, particularly because the game needs the input of the player to become a performance. The MDA framework gives an idea of how the designer can have an influence on player's experience, but in the end, the experience depends on the player herself.

#### **4.3.4. Implications of the Framework**

The first implication of this framework has to do with the role of the player in the

performance. The player is on the side of the aesthetics, since she is the one who experiences the game. The player thus parallels both the audience of the theatre play, and the interactor with software. The player is an active performer because she is also an interactor; but she is also the audience of the performance, since the designers are also creating an experience for her. The look-and-feel of the game, or specific events triggered during gameplay are also part of the experience of the player as spectator of her own interaction, as in the case of juiciness. The “juiciness” of a game (Gabler) is the amplified (and at times excessive) audiovisual feedback that the player receives after every interaction. The interaction can be minimal: hovering over a button may make it sparkle and play a chime. Experimenting with the system of a game to see what happens, is both part of the role of the player as interactor as spectator.

Cut-scenes and micro-narratives, introduced in section 3.6.1.1, can be integrated in the game thanks to this double role of the player. “Juiciness” and cut-scenes can work as rewards for the player doing well in the game, but their purpose goes beyond mere spectacle: they also help the player make sense of gameplay and provide feedback to whether the player is doing well or not.

The framework also explains how videogames can also have a spectatorship, apart from the player (or players) themselves. The player is performing as she is playing the videogame, so other people can watch that performance too. Some games foster an audience more easily because of their ties with other performance forms, such as *Dance Dance Revolution: Ultramix 3*, which has players perform a dance of sorts. *Rock Band* is a band simulator, which attempts to reproduce the experience of being a rock star; although it provides an in-game audience, it also lends itself to being watched by people

who are not playing. Fighting games are also a good example, given their similarities to sports such as boxing, karate, or wrestling.

However, there are other games that bear no obvious relationship to other performance activities and yet they can have a large spectatorship. One of those cases is *Starcraft*, whose tournaments are broadcast on South Korean television. The professional players involved are not only big stars, but also release their best matches on DVD (Evers). Games can also earn a spectatorship when players prove to have great skills, or find exploits in a game. Speed runs (completing a game in the least time possible), or completing a game in the highest difficulty mode are types of gameplay that lend themselves to having an audience. These activities usually find a spectatorship through the distribution of video recordings of gameplay, rather than watching them live.

The audience must understand what the mechanics of the game are, they must be able to make sense of the actions and events in the game, or otherwise they will not watch. The spectatorship of *StarCraft* can be accounted for by the extreme popularity of the game in Korea, which means there is a substantial audience that will understand what professional players do.

Story-driven games also lend themselves to spectatorship, given that the events of the story are enacted on the screen. Again, cut-scenes and micro-narratives help in comprehending the story of the game, integrating the playable sections with non-interactive parts. A spectator can watch someone play *Final Fantasy VII* from beginning to end as if it were a movie—a very long one, with many consecutive fights.

The puzzle-based nature of adventure games, on the other hand, creates a spectatorship different from other genres. One can watch someone else play an adventure

game and follow the story. What usually happens is that the spectator soon becomes involved in the puzzle-solving, giving suggestions and pointers to the person who is playing. Marcel Danesi refers to “the puzzle instinct,” the urge to fill the gap and solve the problem posed by a puzzle (more on this in Chapter 7). Adventure games can be multiplayer without any special design features—their more sedate pace allows for several players to discuss ways to solve the puzzles. This is another example of the dual role of player as performer and spectator, although in this case, the adventure game spectator does not remain one for long.

#### **4.4. Restoration of Behaviors**

Considering videogames a performance activity allows us to understand better the role of the player as performer. The process of restoration of behavior (Schechner, *Between Theater and Antropology*) accounts for how the mechanics of the game shape the behavior of the player.

[S]ome behaviors—organized sequences of events, scripted actions, known texts, scored movements—exist separate from the performers who “do” these behaviors. Because the behavior is separate from those who are behaving, the behavior can be stored, transmitted, manipulated, transformed.” (36-7)

These behaviors are part of the text of the performance, they are types of “scripts” that we mentioned earlier (as opposed to a written text). The process of performance restores those behaviors, if we use Pavis’ term, the performers concretize those behaviors.

An example of this restoration of behavior occurs in The Globe Theatre in London, which is a reconstruction of the theatre where Shakespeare was a writer and

performer. Some of the performances try to reproduce the way the plays would be staged in Shakespeare's time, with no backdrops, lavish costumes, and an all-male cast. Although historical reconstruction is part of the purpose of The Globe, the behaviors that are being restored have also been transformed and updated according to current practices. It is more common to have both male and female actors on the stage than an all-male cast. Groundlings are spectators who watch standing in the courtyard, but while in Shakespeare's time they would only pay a penny and would throw food at the stage if they did not like the play, now groundlings pay five pounds, and behave themselves, even though the groundling tickets are offered as an opportunity to watch plays "as they would have done 400 years ago" (Shakespeare's Globe Theatre).

While the behaviors themselves are part of the mechanics, the restoration of behavior falls into the dynamics of the game. Restoration of behavior in relation to videogames can be more complicated, since there is a range of strategies through which behavior can be restored. This range parallels Juul's distinction between games of progression (having to follow a specific set of steps in a certain order) and games of emergence (finding new behaviors within the constraints of the game system) (*Half-Real*).

On the one extreme, we have games of progression, which pre-suppose an ideal walkthrough. In these games, the player has to find out how to restore the behavior, follow the script that is expected. Music games such as *Rock Band* require that the player hit the notes in synch with the music. There is a definite satisfaction in playing along with a pre-determined pattern, similar to the pleasures of singing in a chorus or participating in a folk dance. In these games, the challenge lies in finding out what the best ways to synchronize are—these are the strategies that players have to come up with.

Adventure games also belong to this type of game, since they usually have just one way to be traversed. The pleasure of restoring behavior in adventure games resides in discovering what that behavior is, by finding the solutions to the puzzles. It is like being an actor in a play that one does not have the script for, and the script is discovered by trying things, exploring the world, and seeing what works. An adventure game is designed to help the player restore the behavior by giving enough pointers and information. The design of the space and the player character are two of the main devices to help the player figure out what to do.

On the other end are sandbox games, where the goals are very broad or the player gets to set her own goals. In these games, there may be types of gameplay that the makers have designed the game for, but players can come up with their own as well. Schechner mentions that the behaviors can also be invented by the performers (*Between Theater and Anthropology* 37), although this does not mean the rules are not in effect any more. Improv theatre, for example, specifies a set of rules before starting each sketch; the actors have to create the performance within those boundaries. Similarly, the pleasure of sandbox games consists of inventing their own goals and how to achieve them, always within the rules of the game. The performance space is also experimentation space: since it is marked as a space separate from everyday life, performers can do things they would not be able to do outside of it.

So far, we have dealt with how videogames compare to other performance activities. But where do stories figure in terms of performance?

#### **4.4.1. Stories in Performance**

It has been established earlier that narrative discourse in games is dramatic,

mimetic. That means that the story is also part of the performance. Again, the comparisons with theatre prove to be quite useful in this context, since theatre is not always necessarily narrative—Balinese theatre, which Artaud so much enjoyed, or dance and music do not have a story, in the same way that not all games incorporate narratives.

In narrative theatre, the story is usually established in the dramatic text. The order of events, in which parts of the text are included or added also correspond to the text of the play. Even if the modifications to the original text are not made by the author, but by the actors or the director of the performance, those changes usually occur before the performance takes place—the text, in the end, is pre-set.

The performance framework facilitates understanding how stories can be part of videogames as well. As Figure 7 illustrates, each column of the framework can relate to stories in a different way, resulting in different types of stories: embedded stories, emergent stories, and retelling of gameplay.<sup>18</sup> The terms below describe the possible relationships between games and stories. Depending on the game, we can find all three types present, only two, or none at all.

---

<sup>18</sup> I am adopting the terms emergent and embedded again from Marc LeBlanc, as referred to by Salen & Zimmerman in *Rules of Play*; Costikyan also uses these terms to define the different types of relationship between stories and gameplay (“Games, Storytelling and Breaking the String”).

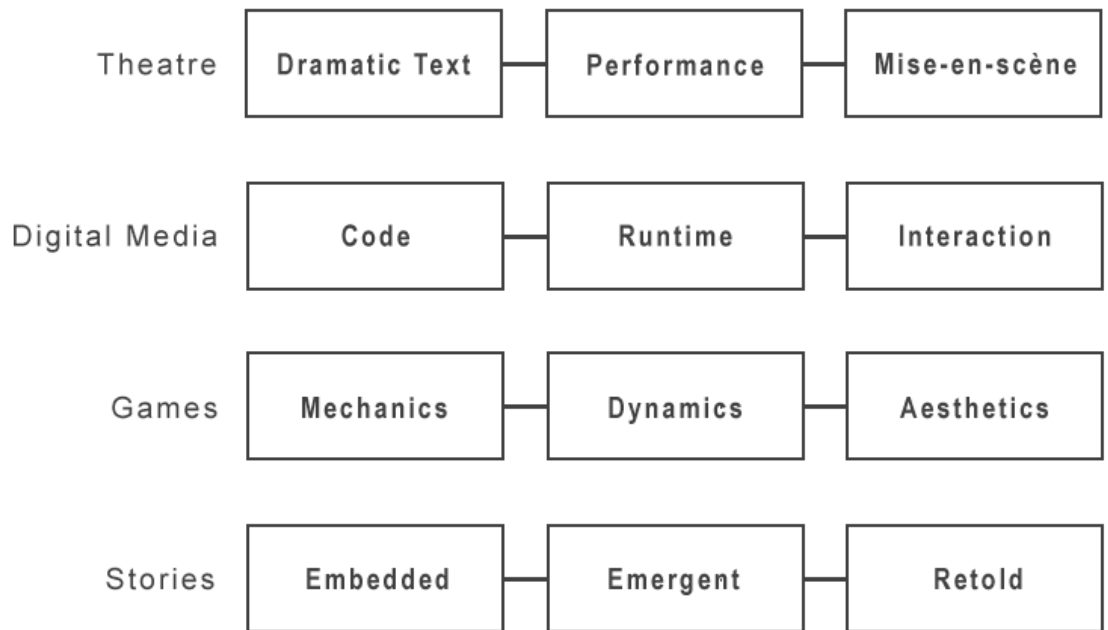


Figure 7: Stories in relation to the Performance Framework for Videogames.

#### 4.4.1.1. Embedded Stories

Stories can be prescribed by the mechanics, just as by the dramatic text. There is a predetermined story that will take place as the player progresses in the game. Embedded stories usually imply a stronger authorial voice on the part of the game makers, since they are pre-set and unchangeable. However, these stories are also *in potentia*—they need the interaction of the player to actually take place, to be performed.

Embedded stories relate both to the pre-history of the game and the player's history. In the case of the pre-history of the game, the embedded story is whatever has happened in the world before. In some games, part of the gameplay consists of discovering that pre-history, following the model of detective novels described by Todorov (McQuillan 120-127).



Player's history can also be defined by embedded stories, particularly in the case of story-driven games. The "ideal walkthrough" is an embedded story, pre-determined by the designers. If there are multiple endings, those endings are usually pre-scripted. The sections of the game where the player stops interacting with the game, and the story is represented using other narrative forms, from written text to comics or cut-scenes, are also part of the embedded story of the game.

As Salen and Zimmerman define them, embedded stories "provide motivation for the events and actions of the game, players experience embedded narrative as a story content" (*Rules of Play* 383). The story can also drive the gameplay—the player wants to know what happens next, so she continues playing. Embedded stories are ideally at the service of understanding the rules and goals of the game, and providing the fictional basis for its mechanics. For example, the detective story framework of games such as *Deadline* or *Gabriel Knight: Sins of the Fathers* provides the basis for a game where the core mechanics are gathering information and cross-questioning witnesses.

Embedded stories are predominant in adventure games, since there is a strong story component dictating what the gameplay should be. Every step to make the story unfold is pre-determined, since puzzles usually only have one way to be solved. The player has to discover what that story is, but does not get to alter it or create her own story.

#### 4.4.1.2. Emergent Stories

Emergent stories are the events that take place during gameplay, as the result of the player interacting with the rules of the game. They are the story of the performance itself, the result of the dynamics of the game. This type of story falls exclusively into the

realm of the player's history, since they only refer to what happens during gameplay that has not been pre-scribed by the game makers. The events that take place during emergent gameplay become part of player's history.

Simulation games such as *The Sims* or *Civilization III* are examples that promote this type of this story generation. These games are rich models of very specific real-world systems: living in the north-American suburbs, politico/techno/social history of a region, or a movie studio. Simulation games also have their limitations, although the more complex and nuanced the system is, the longer it takes for it to get repetitive. The myth of the Holodeck (Murray, *Hamlet on the Holodeck*) is the quintessential example of emergent storytelling—a computer system rich enough to produce different stories every time someone interacts with it. One of the problems of pursuing a Holodeck-like model is that the system would have to be extremely complex, and would have to be able to anticipate every single action the interactor may try.

Warren Spector refers to a “shared authorship” model, which combines embedded storytelling with emergent storytelling, as one of pursuits of current game development. However, adventure games evidence that this combination is problematic and difficult. As we will see in the following chapters, adventure games are simulations, but since its player's history is mostly embedded, there is not much opportunity for the simulation to generate story events. Gameplay may change the order of events, and the player may discover more or less about the world through exploration, but in the end adventure games have little room for emergence. Current game development can thus learn from the issues that have arisen through the evolution of the adventure game genre, to avoid the same pitfalls and hopefully seek new avenues of innovation to create new shared

authorship models.

#### 4.4.1.3. Retelling of Gameplay

Retelling refers to the experience of the player, both as an interactor and as audience, as expressed in the form of a story. These stories can be a retelling of gameplay, what a player can tell after playing (or while playing, in some cases). Retold stories belong to the third column of the framework, which refers to the *mise-en-scène*, and to the aesthetics of the game. Retelling includes all the other stories of the game, both embedded and emergent.

The retelling of game events has been a recurring argument both for pointing out the narrative nature of games (Murray, *Hamlet on the Holodeck*; “From Game-Story to Cyberdrama; Salen and Zimmerman, *Rules of Play*) and for disputing it (Juul, “Games Telling Stories?”; Eskelinen, “Towards Computer Game Studies”). Branigan argues that narrative structures are a natural cognitive schema, and that the human mind uses narrative structures to understand events, even if those events are not necessarily intended to be a story. As we saw, however, Branigan’s argument, is so all-encompassing it makes it difficult to define narrative at all.

Based on Branigan’s approach, we could also discount the importance of game design as part of the creation of the story that will be retold. We have already argued that the designer does not have direct control over what the player may do, but game design can determine the events that can happen in the game. By emphasizing the retold story over embedded and emergent stories, we would leave out how game design partakes of the other types of stories in relation to the performance. The intent of this dissertation is precisely the opposite: investigate how game design relates to and shapes the stories of

the game.

The recurring example of trying to re-tell what happens in *Tetris* as a story also challenges Branigan's argument, because even if it is possible to tell the events, they do not make for a very interesting story (Spector). The reason is the weak sense of causality from one event to another, which is an important factor in defining what a story is.

Therefore, games where the causality between events is clear to the player lend themselves to retelling better than others. This is the case of story-driven games, where the story is predominantly embedded, and emergent storytelling, where the player can see how her choices create a chain of events. The game itself can support retelling as well, from screen captures to make albums (*The Sims 2*) to video capture (e.g. flight simulators such as *Secret Weapons of the Luftwaffe*). These tools are not part of the mechanics of the game, they are a complement that can be used for storytelling outside of the game to produce web photo albums or videos, a.k.a. machinema.

Adventure games also lend themselves to being retold, either by reproducing the embedded story of the game (which becomes the player's history), or by telling what the player tried in order to solve a puzzle. Game walkthroughs of adventure games, although they could take a story form, frequently limit themselves to being a list of the commands needed to complete the game.

#### **4.5.Conclusion**

This chapter has dealt with the dramatic discourse of videogames, as the "how" stories are told, and introduced a framework to understand games as performance. These foundational concepts help us understand not only how videogames are a performative medium, but also have helped us identify the particularities of adventure games with

respect to other game genres. In the following chapters, we will see how performance serves as the link between game and story in adventure games. In particular, the performance of the player with the simulation involves interacting with the fictional world according to the rules of the game, thus making the events of the story happen through the interaction.

## CHAPTER 5 ADVENTURE GAMES ARE SIMULATIONS

### 5.1.A Story-Game Hybrid

The goal of this chapter is to prove that adventure games are simulations, contrary to the extended belief that they are linear interactive stories. Espen Aarseth, refers to them as “story-game hybrids,” as opposed to other games which he considers simulations (“Genre Trouble” 51). Since the player is supposed to complete a story as she plays, and represented content predominates during gameplay, it may be easy to overlook how they simulate a fictional world. However, their hybrid nature is precisely what makes adventure games worth our attention, because of how story shapes the performance. In this chapter, we will discuss how the fictional world and the rules intersect to create the simulation, how the players perform in it, and how this model applies to adventure games. The fictional world of adventure games is implemented through the rules of the game as a simulation, where all the objects and characters must respond to whatever the player tries to do (even if it is to tell the player that the action failed).

Stating that adventure games are simulation is not new. In what was probably the first paper written on adventure games, Lebling, Blank and Anderson referred to both *Zork* and *Adventure* as a “Computer Fantasy Simulations”, and discuss it in terms of its implementation. Aarseth himself (perhaps not inadvertently) talks about the origins of the genre as a simulation: “[...] in 1976, [...] Donald Woods turned William Crowther’s text-based cave simulation into a fantasy game” (“Genre Trouble” 51).

What distinguishes adventure games from other genres is that the goals of the

game, rather than be set by the rules, are set by the story. These goals are also concatenated, establishing causal relations between the actions of the player in the game. The contrast between the simulation and the progression in the game is the result of having a story-driven game.

## 5.2.Rules

“Simulation” has already been introduced as the intersection of the fictional world with the rules of the game. Rules are also one of the defining qualities of performance activities; by extension, they are fundamental to define a game (Huizinga; Caillois; Salen and Zimmerman; Juul *Half-Real*). Before discussing the simulation at length, let us discuss how the concept of rules applies to videogames, and how they determine the performance.

Rules dictate what can and cannot be done in the game, creating a system with which the player interacts.

“Rules specify *limitations* and *affordances*. They prohibit players from performing actions such as making jewelry out of dice, but they also add meaning to the allowed actions and this affords players meaningful actions that were not otherwise available; rules give games *structure*.”<sup>19</sup> (Juul 58)

Thus game rules limit what the player can do, e.g. in football the ball cannot go outside of the field boundaries, and players cannot touch the ball with their arms or hands. Rules also enable the player to do actions that will have a meaningful effect on the game state by setting up goals: polishing a chess piece is not prohibited, but it does not have effect on the game. Moving the knight to capture a bishop, on the other hand, is

---

<sup>19</sup> Emphasis in the original.

made possible by the rules, and affects the game state. Winning the game means achieving the final goal of the game.

As we stated in the previous chapter, the rules of the game are parallel to the dramatic text in theatre. The rules, like the dramatic text, can change when the game is played: the restoration of behavior can interpret and change those rules depending on the goal of the performance. In non-digital games, these transformations can take the shape of house-rules or implicit social rules (e.g. a parent letting their child win). In videogames, the rules are enforced by the computer code (they belong in the same column of the performance framework), although players may also add their own goals. *The Sims* is a good example of how to let players establish their own goals, the rules of the game are basically at the service of the player to attain those goals.

How the goals are set is one of the features that distinguishes adventure games from other genres. While the rules establish how the game plays (more on that below), the goals are not set by the rules, but by the story. Goals set by the rules can be achieving the highest score possible, or finish the race first; while goals set by the story can consist of finding information, e.g. solve a murder case (*Deadline*, *Gabriel Knight: Sins of the Fathers*), or discover the hideout of the Order of the Templars (*Broken Sword: Shadow of the Templars*). Achieving a game goal set by the story can also consist of achieving a specific state of affairs in the fictional world, e.g. become a pirate (*The Secret of Monkey Island*); rescue the royal family who has been kidnapped by a wizard (*King's Quest V*).

### **5.2.1. Rules vs Mechanics**

Videogames can make a distinction between types of rules. Rules in general



establish the relationships between the different game elements (Järvinen 30).<sup>20</sup> For example, in *Super Mario 64*, some game entities are enemies and will hurt Mario (the Goombas), while some objects help Mario obtain special abilities (power-ups in the form of different colored caps).

Rules also establish how the player interacts with the game, shaping the agency of the player. The rules that provide players with agency in the game are the game mechanics (Järvinen 250-274; Sicart). The game mechanics are a subset of the overall game rules. They establish, for example, how the player character moves in the world, how to acquire objects, how to interact with non-player characters. More importantly, they also set up the consequences of the interaction in the world—mechanics have an effect on the game state, so the player knows what to expect after performing a certain action.

An easy way to identify the mechanics of the game is by listing the list of verbs that describe the actions of the player (Järvinen 263). The set of verbs in a list can help us identify the type of game. For examples, in First Person Shooters there is walk, run, shoot, change (weapon), reload, strafe; in racing games there is run, brake, accelerate, turn, drift. In adventure games, verbs can be as explicit as having to type them, as in the case of text adventure games. The verb list can be made explicit either by displaying a menu, or by listing the verbs in the manual or a tutorial. Throughout this dissertation, the term “verb” will be synonymous with “player action.”

This distinction between rules and mechanics helps us understand the relationship

---

<sup>20</sup> Järvinen (55) identifies game elements as systemic elements (components, environment), compound elements (ruleset, game mechanics, theme, interface, information), and behavioural elements (players, contexts). For the purposes of this thesis, the elements of videogames are the basic qualities of performance: time, space, objects (including characters).

of computer and player as performers. The overall rules are followed by the computer as performer, whereas the game mechanics are the ones available to the player as performer.

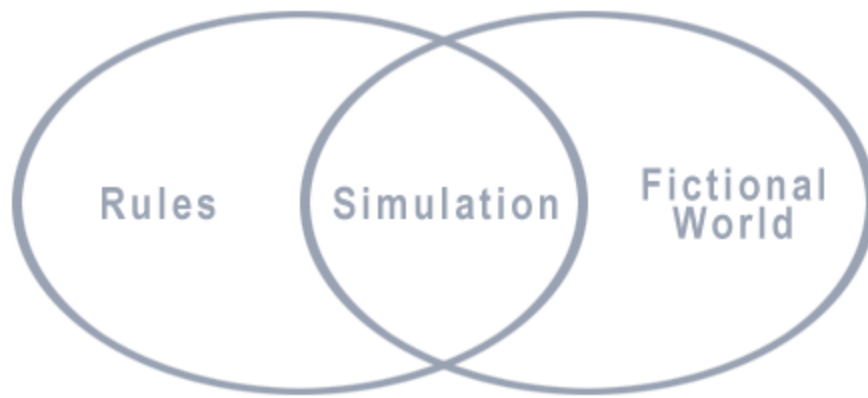
As Sicart explains:

Game mechanics are concerned with the actual interaction with the game state, while rules provide the possibility space where that interaction is possible, regulating as well the transition between states. In this sense, rules are modeled after agency, while mechanics are modeled for agency. (Sicart)

Rules thus create the potential for interaction with the game, whereas mechanics establish how the interaction take place. In either case, as Schechner said (*Performance Theory* 12-13), rules dictate how the performance is going to take place and what its proper form is, for both the player and the computer as performers. The distinction between rules and mechanics will also help us understand what are the aspects of the performance that take part in generating the events of the story, as we will see in the following section.

### **5.3. Defining Simulation**

The term “simulation” is recurrent in game studies, although there is a range of reasons why it is used. The relationship between games and simulations is older than what we now call “game studies”. Simulation games have been used as a tool for academic research since the 1950s, in fields as diverse as social studies, education, international relations, and natural resources (Klabbers 17). These games usually modeled a real-life system, such as social environments in a company, a city, or a specific ecological environment. The multidisciplinary applications of simulations have been used as a way of predicting and studying the behaviors of those systems.



*Figure 8: Simulation is the intersection between the fictional world and the rules.*

The term simulation has acquired a different meaning in videogames, particularly in the context of fictional worlds. The presence of a fictional world seems to be a prerequisite to have a simulation. According to Juul, a simulation is “the implementation of a fictional world in the rules of a game” (*Half-Real* 170). For our purposes, we are going to define simulation as the intersection of the rules of the game with the fictional world, as illustrated in Figure 8. In that particular intersection, the rules of the game are implementing a specific part of the fictional world. The simulation is the system that recreates the fictional world with which the player interacts.

### **5.3.1. The Struggle between Simulation and Story**

Aarseth and by Frasca are two of the main proponents of the study of games as simulations, practically equating both terms, in order to distinguish them from narratives. Aarseth affirms that “the computer game is the art of simulation,” although he does not quite get around to provide a definition of the term.

Simulation is the hermeneutic Other of narratives; the alternative mode of discourse, bottom up and emergent where stories are top-down and preplanned. In simulations, knowledge and experience is created by the player's actions and strategies, rather than recreated by a writer or moviemaker. ("Genre Trouble" 52)

By understanding simulation as an hermeneutic, Aarseth highlights the player's understanding of the fictional world as part of what distinguishes games from other media, even more so than interaction. This definition of simulation seems compatible with the tenets established by our performance framework—the player making sense of what goes on in the performance is key for the activity to be considered performative.

Frasca defines simulation as "the modeling of a dynamic system through another system" ("Videogames of the Oppressed" 86). Frasca admits that his definition is far broader than the one used in Computer Science, and by extension than the one used in simulation games: "Traditionally, simulations model real systems and connote an intention of scientific understanding. When I use the term it is in order to describe a different form of representation and, as in modern semiotics, I do not see the need for a real referent."<sup>21</sup> Frasca is thus using the term 'simulation' as an alternative to 'representation,' which harkens back to traditional narrative ("Simulation 101"). The choice of terms intends to highlight the dynamic nature of simulations, whose outcomes are variable, as opposed to the fixed and unchanging nature of narrative ("Videogames of the Oppressed" 86). Both Frasca and Aarseth aim at establishing the difference between games and narrative, rather than as an acknowledgement of the pre-existing academic

---

<sup>21</sup> The lack of a "real referent" allows us to account for games that model systems that are not reproducible in the real world, such as space flight simulators (*X-Wing*), or fictional sports (*Harry Potter: Quidditch World Cup*). This is not incompatible with our definition simulation: the worlds that are being simulated are fictional, imaginary. The potential absence of a real-world system that is being modeled seems to be closer to the concept of 'simulacra' as described by Baudrillard.

field of simulation games. In Frasca's case, highlighting 'simulation' is a way to emphasize the systemic nature of games, since they are based on a set of rules.

Simulations are indeed different from narratives. The argument here is that one of the ways to reconcile them is through performance in the simulation. By interacting with the simulation, the player is restoring a behavior; that restoration (re)creates the story of the game. A change in the game state is also a change of state in the fictional world, and therefore corresponds to an event, as Chatman defined it (19). Not all changes in the game state may correspond to an event; for example, clearing four pieces in *Bejewelled* changes the game state but it is not a story event. In order to consider a game state change an event, it has to take place in the fictional world and involve the existents of the story.

The participatory qualities of the simulation also allow the player knowing more about the fictional world of the game: "Simulations allow us to test their limits, comprehend causalities, establish strategies, and effect changes, in ways clearly denied us by fictions, but quite like in reality" (Aarseth, "Doors and Perception"). Thus by interacting with the fictional world, the player can figure out its rules, experimenting with what the simulation allows.

Defining simulation as the intersection between the fictional world and the rules of the game helps clarify the misconception about adventure games not being simulations. Aarseth dismisses adventure games as games, because they are closer to stories and do not seem to have a dynamic structure based on rules. Aarseth also seems to mistake the way the fictional world is represented with its simulation:

In the early text adventure games, the fictive layer often dominated, with fixed descriptions that changed very little or not at all. With today's increasingly more physics-heavy 3D games, the drive away from fiction towards simulation continues with the development of dedicated physics processors (PPUs) in order

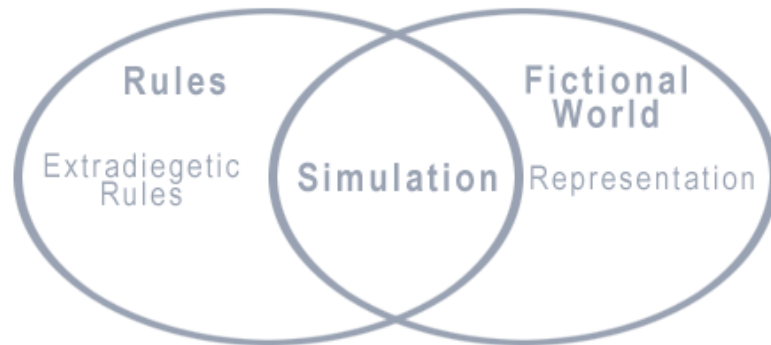
to emulate real-world physics ever more faithfully. (Aarseth, Doors and Perception)

Since “fiction” for Aarseth implies “non-interactive”, he equates the textual descriptions of the world as shutting out the participation of the player. That assumption is somewhat misguided. Adventure games simulate the existents of the fictional world: navigable space, characters and manipulable objects. Most adventure games are implemented using some variety of Object Oriented Programming, so that objects in the world are usually programmatic objects as well. Game objects have an action or set of actions attached to them, which determine what the player can or cannot do with them; they frequently have a description attached to them as well. The fictional world is thus programmatically simulated, creating a virtual space where the player can perform with objects that become part of that performance.

Given that they are programmatic simulations, one could model a physics system in a text adventure, for instance. There are IF pieces which are actually shooters—a text version of *Quake* and *Gun Mute*, a text shooter set in the Wild West. Adventure games have rules, which implement how the world works; goals, which are attained by puzzle-solving; and mechanics, which are the actions the player performs in order to solve the puzzles and explore the simulated world.

The representation does not preclude the simulation either. As Montfort succinctly puts it, in the context of Interactive Fiction “there may be things that are narrated during an interaction but are not simulated” (“Generating Narrative Narration in Interactive Fiction” 29). The following section explains the relationship between the representation and the simulation of the fictional world.

### 5.3.2. Rules-Simulation-Fictional World



*Figure 9: Rules and the fictional world which are not part of the simulation.*

In order to clarify how adventure games are simulations, the relations between the rules of the game, the simulation, and the fictional world must be defined. Figure 8 is the reference diagram to explain their relationship. Where the rules and the fictional world intersect, there is the simulation. The simulation is the implementation of the fictional world as a system. For example, the simulation of *Grand Theft Auto: Vice City* is a large city where the player can hijack cars and drive around, shoot at people, and run away from the cops. The day / night cycle of the game, the drivers, pedestrians, and policemen, are all functional, they behave in accordance with the rules—they are simulated. On the other hand, the player cannot interact with the buildings—they stage the action, the player can drive around them, but only a few can be entered. The entrance and interiors of most buildings are therefore not simulated. The buildings are represented buildings but are not sheer décor—they help the player situate herself in the fictional world, they help understand the where she is in the city. The representation is part of the fictional world, and it helps the player make sense of the game. Thus, the representation is part of the

fictional world, but not the simulation (see Figure 9).

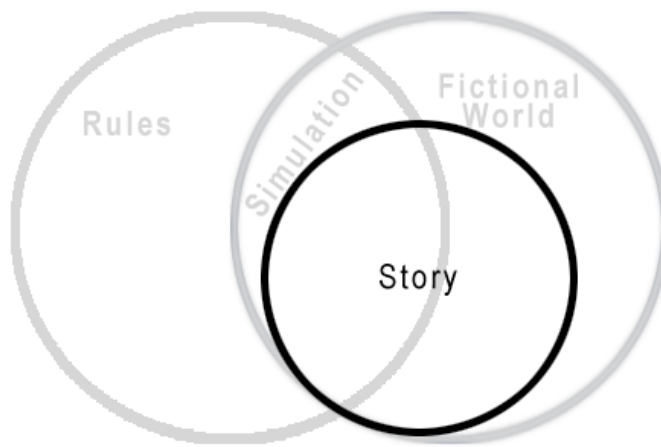
Conversely, there are rules that are not part of the simulation, although they still regulate the performance of the player in the game. In the same way that there are aspects of the fictional world that are not simulated, but represented, there are rules of the game that are not part of the simulation: they are extradiegetic rules (see Figure 9). These rules, since they are not part of the world, refer to the aspects of the game that are not part of the fictional world. The save and load options of the game, for example, are extradiegetic. Continuing with *Grand Theft Auto: Vice City*, the score is extradiegetic, determined by the rules of the game. The score reflects the success of the actions in the simulation, but it is not part of the fictional world. So is the number of stars displayed on the graphical user interface, which indicate the level of alarm that the player's actions have created. A single star highlighted means that one policeman is chasing the player character, whereas when all six are lit up, all the police force in the city plus the military are trying to hunt him down. Again, although not part of the fictional world, these elements are providing information about the game state.

The distinction between rules-simulation-fictional world is also a way to account for the more general term "gameworld." "Gameworld" has been used to refer to the world of the game in general, without making the distinction whether it is implemented in the rules or the player can interact with it. Gameworld is a term commonly accepted in game studies, since it refers to the world of the game. In the case of adventure games, making the distinction between simulation and fictional world will help us differentiate throughout the dissertation between the world where the story takes place, and the parts of the game where the player can actually affect and interact with that world.



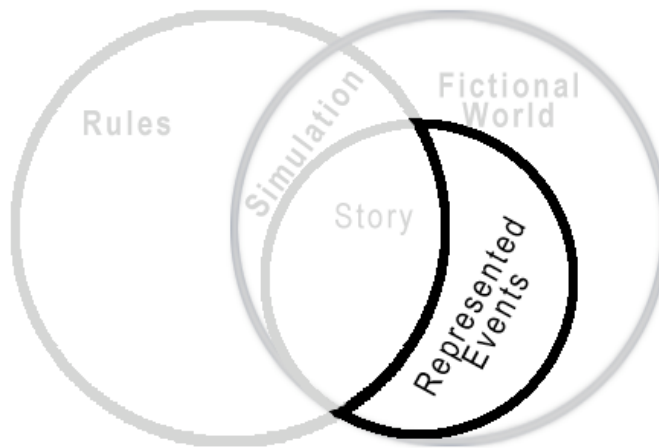
### 5.3.3. Integrating the Story in the Simulation

How does the story integrate in the simulation? The fictional world is the realm of the story, and therefore the events of the story must take place within it (see Figure 10). The events of the story can be generated by the simulation, or they can be represented, e.g. as narration, as a cut-scene, as a series of illustrations.



*Figure 10: The story in the game is part of the fictional world.*

Not all aspects of the fictional world may be part of the events of the story of the game. As we saw in *Super Mario 64*, we do not know who the king or queen of the realm are, nor what the economy is like. In the same way that some events of the story are not represented, there may be existents of the story that are not simulated, and yet they are part of the story. For example there may be characters who are talked about but not encountered in the game, as is the case of Mr. Robner, the victim of the murder case of *Deadline*. These characters, like the buildings in *Grand Theft Auto: Vice City*, are the parts of the fictional world that are represented but are not simulated (see Figure 11).



*Figure 11: Events and existents that are not part of the simulation are represented.*

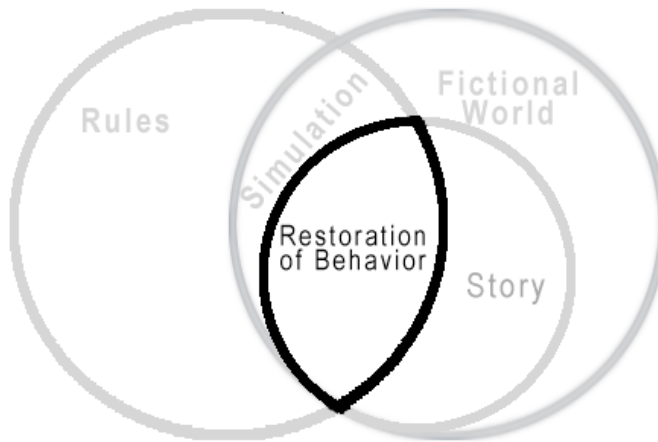
The representation is thus part of the game and the process of making sense of the world. However, the representation alone does not make a videogame—the foundation of the game is the simulation, the story is a means to understand that fictional world better. As Aarseth puts it:

Stories and simulations are not totally incompatible, but the simulation, as a primary phenomenon, must form the basis of any combination of the two, and not vice versa, just as with stories and life. When you have built a simulation, such as a rule-based gameworld, you may use it to tell stories in (or for other purposes) [...]. (Aarseth, “Genre Trouble” 52)

Thus the simulation of the fictional world is the basis for the story, rather than the other way around. The simulation can help generate the events of the story as the player restores behavior, either the events of the embedded story of the game, or emergent stories generated by the system of the game.

Restoring behavior in the game is playing the game the way it is supposed to be

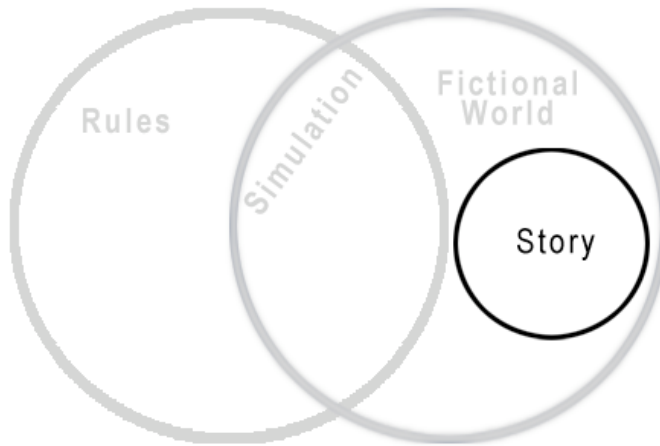
played, restoring the performance to its proper form. As we saw, games of emergence and games of progression restore behaviors differently: the first can generate and improvise behaviors based on the rules, while the second have to restore behavior in a specific, usually prescribed way. The behavior can be restored without those actions producing a story—playing *Rock Band* is an example of a non-narrative restoration of behavior. In story-driven games, the behavior restored by the player generates the events of the story, the events that are simulated and are not represented (see Figure 12).



*Figure 12: The restoration of behavior generates the story events through interaction with the simulation.*

There are games where the story and the simulation does not overlap. This is the case of *The 7<sup>th</sup> Guest*, where the story is only represented, and does not provide information to the player that helps her solve the puzzles. It is still a basic simulation, since it presents a 3D navigable space, where a series of puzzles are distributed. This is

further evidence that *The 7<sup>th</sup> Guest* is not an adventure game, since adventure games follow the model presented in Figure 10, whereas the model of *The 7<sup>th</sup> Guest* is shown in Figure 13 below.



*Figure 13: The integration of the story in The 7th Guest*

As we will see in the following chapter, the puzzles are part of the simulation, since they are both regulated and they are part of the fictional world. Since puzzle-solving constitutes the core gameplay, the restoration of behavior is based on puzzle solving. This does not mean that it is the only thing the players can do—the game mechanics should also allow exploring the workings of the simulation, which at the same time helps the player learn more about the fictional world. The exploration of space and actions is one of the distinctive features of adventure games, as we saw. This feature would not be possible if adventure games were not simulations.

#### 5.4. Abstracting the Fictional World

Fictional worlds are incomplete, as we saw in Chapter 3; similarly, the simulation does not implement the whole world, it only incorporates the aspects of the fictional world that are relevant to gameplay. It is not only that some parts of the fictional world and others are simulated, it is that what is not simulated is usually not relevant to the interaction. For example, when running on the racing tracks of *Wipeout*, the player can see there is a larger fictional world, there are futuristic buildings and advertisements, but there is no information about the society or culture where those races take place. The expected performance in simulation of *Wipeout* is racing. The simulation is designed to host a specific type of performance activity, and that is what may delimit how big and elaborate it is—there must be a selective process before turning the world into a simulation. Adventure games were one of the first genres to design worlds purposely, i.e. for the player to complete a specific goal or set of goals. As David Lebling and Mark Blank, two of the original implementors of *Zork*, noted:

Obviously, no small computer program can encompass the entire universe. What it can do, however, is simulate enough of the universe to appear more intelligent than it really is. This is a successful strategy only because CFS games [Computer Fantasy Simulations, their term to refer to what would become adventure games] are goal-directed. As a consequence, most players try to do only a small subset of the things they might choose to do with an object if they really had one in their possession. (Lebling, Blank & Anderson 52)

Goals thus become a means to indicate to the player the kind of performance that is to be restored in the game. Establishing the goals of the game clearly is a way to guide the player to the right way to play the game, in order to maintain the proper form of gameplay.

Story-driven games tend to present more fleshed-out fictional worlds than other genres. We can compare the cities of *Wipeout*, or the different stages of *Super Mario 64*, with Tolkien's universe in *The Hobbit* graphical text adventure, or the space-opera galaxies of the *Space Quest* series. Having worlds that are more complex and elaborate also means that there are potentially more elements that can be simulated, and also that there will be more aspects that cannot be implemented. Rather than tackling those gaps as a problem, games can take them as an advantage, since they invite the player to fill them in and explore the world.

Establishing clear goals does not prevent players from attempting to do other things in the game that may not have anything to do with that goal, especially if the goal is not clear to the player. Adventure games require the player to experiment and explore the game, so even if the goal is given, the player is expected to poke and probe the simulation to find out how it works. The player thus can expose the limitations of the simulation.

In order to cover up those limitations, the rules have to anticipate what players may try to do. That can take the form of associating the right actions to each specific object in the game. A simulated wooden chair would include sitting on it, standing on it, or pushing it around (or not, if it is a heavy chair), but it cannot be put inside a pocket or eaten. Other more specific behaviors, such as taking it apart and turning it into firewood should also be predicted by the rules.

Designing a world with this level of detail takes a lot of work on the game designer's part, and it is not the most efficient way to simulate the fictional world—there can always be something that a player can try and will not be anticipated by the rules. A

strategy to prevent implementing a whole world is selecting what aspects of the fictional world will be most relevant to gameplay.

#### **5.4.1. Levels of Abstraction**

In order to implement the fictional world, it has to be abstracted first to the components essential to the game and made into a system. Abstraction is a primordial step in the design of the simulation (Gingold 14-5), it is a selection process that determines what is essential to allow the player to interact with. As Lebling, Blank and Anderson affirmed in the quote above, “no small computer program can encompass the entire universe” (52). Thus in implementing the fictional world, the gamemakers have to select what is essential to the player to restore the behavior associated with the game.

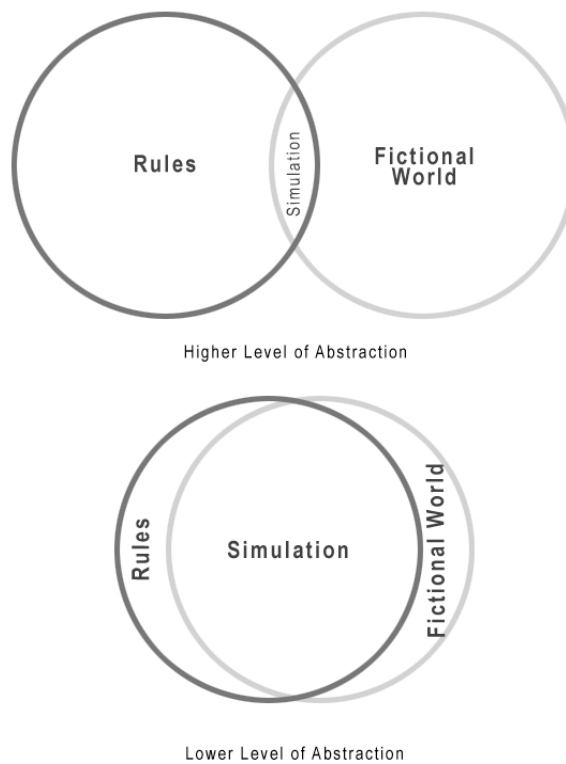
The concept of “level of abstraction” is commonly used in game design to refer to how a specific model is implemented as a game system (see Gingold; Will Wright in Pearce, “Sims, Battlebots, Cellular Automata God and Go”). Juul refers to the level of abstraction of a videogame as the border between the fictional world and how it is implemented in the rules, within which the player can act.

[T]he player can only act on a certain level, outside which the world is either crudely implemented [...], simply represented [...], or simply absent [...]. (“A Certain Level of Abstraction” 510)

Juul talks about how *Cooking Mama* presents a kitchen as a fictional world, but the player’s interaction cannot be that of a real kitchen: cucumbers can only be cut in a specific way, and one cannot order take out instead of cooking (510). The parts of the game that are not relevant to gameplay have less functionality (e.g. you see a picture of the dish you cooked, but there is no action to eat it), or are not implemented at all (e.g.

the player cannot visit the rest of the house where the kitchen is). Following this logic, the events in the fictional world that cannot take place as a result of the game system will either be represented or left out.

For the purposes of this thesis, rather than interpreting the level of abstraction as a border we will interpret it as an area, as the amount of overlap between the rules and the fictional world. This illustrates more clearly the different degrees of this abstraction. A smaller overlap, where either the rules or the fictional world present a larger area in the diagram, indicates a higher level of abstraction, and a less fine grained interaction. The larger the area of overlap between rules and the fictional world, the lower the level of abstraction, and the more high-fidelity the simulation is (see Figure 14).



*Figure 14: Level of Abstraction is the overlap between the rules and the fictional world.*



Throughout the thesis, the term “abstracted” will be used as a quantifiable adjective, to refer to this implementation of the fictional world in the game mechanics, rather than to its traditional meaning of either “theoretical” or “non-representational.”

As an illustration of how different levels of abstraction can affect the design of a game, we can compare *Pong* with *Virtua Tennis*. *Pong* can be considered a highly abstract version of tennis, although it is so abstracted that it could also be ping-pong or badminton. The player cannot get closer to the net, since the racquet can only move up and down and it does not swing. There is no serving either—after a point has been scored, the ball re-spawns from the middle of the screen, towards the side of player who just lost. On the other hand, the level of abstraction of *Virtua Tennis* is relatively low. The game is three-dimensional, and there are a variety of ways to hit the ball which involve not only the angle but the strength and height of the shot. Even the choice of court and its surface will affect how the ball bounces. Both *Pong* and *Virtua Tennis* are simulations, and the difference is not only that one has big, blocky, black-and-white graphics and the other has 3D-rendered graphics in color. *Pong* has implemented fewer rules of the game of tennis, so few that it is difficult to identify what its fictional world is, whereas *Virtua Tennis* is simulating a tennis tournament, including the likeness of professional players and complex physics.

Level of abstraction refers to the level of detail of the system, independently of its visual representation. Adventure games are a clear example: the lavish backgrounds of *Myst* (1993) were beautifully rendered graphics when the game was released, but the functionality of the world was much more limited than the one in *King's Quest: Quest for the Crown* (1984).

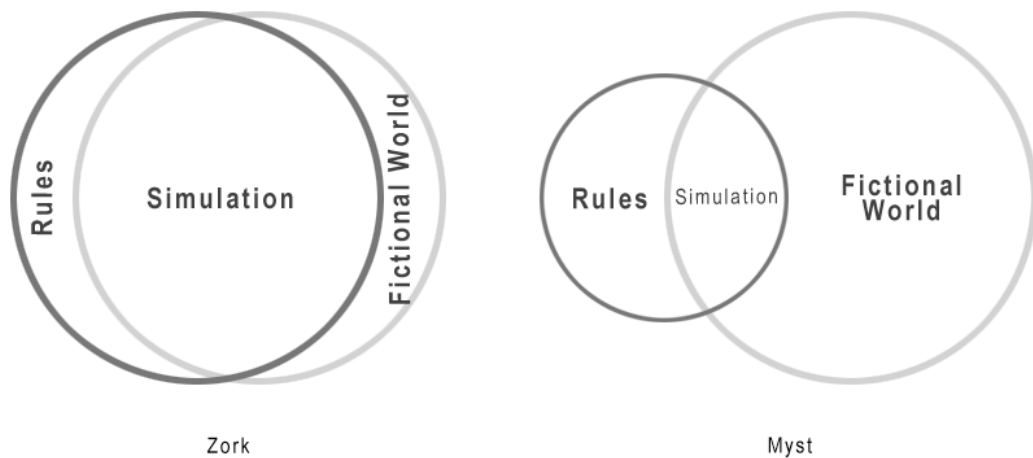
### 5.5. The Simulations of Adventure Games

The concept of levels of abstraction is useful to understand how the fictional world is implemented in the rules. It is particularly relevant to adventure games, because the levels of abstraction have changed historically, from lower levels of abstraction in text adventures to higher levels of abstraction in the latest point-and-click adventure games.

The difference between the simulated worlds of adventure games and the simulations of other genres comes from the historical evolution of the genre. Whereas in other videogame genres the technological advances have allowed the creation of more complex worlds, with lower levels of abstraction (e.g. more nuanced physics systems, more complex behaviors of the game entities), the simulations of adventure games have steadily adopted higher levels of abstraction. Better computers have made adventure games look better and better—one of the attractions of *Myst* are precisely its beautiful (for its time) graphics, for example. However, the simulation of the *Myst* world, how it actually works, uses a higher level of abstraction—there are less items to interact with, and less actions to perform in the simulation. The player character can only take one object at a time, and that is a book page. This forces the player to choose between two pages in each area (Age) to bring back to the island of *Myst*. This restriction on the type of objects that the player can pick up makes clear that the player does not have to hoard objects in the world. Thus, picking up a page in *Myst* becomes non-trivial, since it involves making a choice of what page to bring.

*Zork*, on the other hand, represented its world textually, but its world used a lower level of abstraction. Most objects in the game could be picked up and dropped, the

player can try to burn objects and then put out the fire with water to see what happens. The simulation of the objects in the world is more fine-grained, and the mechanics of the game also offer more nuanced actions to the player. In short, *Zork* uses a lower level of abstraction to construct its simulation, while in the simulation of *Myst* the rule system is more simplified, thus creating a more limited simulation and offering the player less variety in the available actions (Figure 15).



*Figure 15: Levels of abstraction of Zork vs. Myst.*

The higher level of abstraction in adventure games also means that the events of the story have less room in the simulation, so they have to be represented. If there are fewer possible actions for the player, but there is still a story to unfold, the events have to take place in the representations (again, cut-scenes, descriptions), curtailing the player's agency over the events.

### 5.5.1. Different Levels of Abstraction Reflected in the Objects

With respect to the objects in the world, a lower level of abstraction means that the object has a more detailed functionality. This also means that the player can try more actions involving that object, and that those actions will very likely receive a response, even if the action is not part of solving a puzzle. On the other hand, a higher level of abstraction means that an object will only respond to the specific action that makes it relevant to solving a puzzle.

For example, both *The Lurking Horror* (a text adventure) and *Maniac Mansion* (graphic adventure) feature a microwave oven. A microwave is not a versatile object that can be used repeatedly in a game, but its inclusion makes the fictional world richer. In *The Lurking Horror* all the buttons of the oven are operational: it can be opened/closed, the player can put food inside to cook it (Chinese food in this case). In order to cook, the player has to press the right buttons: enter the time using the numeric pad (press 5, 0, 0 which means five minutes), set the level (high) and then start the oven. On the other hand, the microwave oven in *Maniac Mansion* can be opened/closed and turned on/off. The player character will refuse to use it if it is empty, and there are only two objects that can be used in the oven.<sup>22</sup>

The functionality of the microwave in *Maniac Mansion* is less fine-grained than the one in *The Lurking Horror*, since there is a smaller range of actions that apply to the specific object. The more restricted the functionality of an object is, the less verbs can be applied successfully to it. This difference between levels of abstraction presents an

---

<sup>22</sup> One is a hamster, who obviously does not survive the plight, and the other is a glass jar full of radioactive water. Either of these actions kills the character who performs it.

interesting dilemma. Being able to do more things with an object means that it will be more difficult for the player to find out what to do with it, so the player will be more prone to error. On the other hand, a narrower range of actions means that the simulation will also be less rich, making the world less interesting to explore, more restricted and more obviously artificial.

Heating food in the microwave is a key puzzle to advance in *The Lurking Horror*, but since it requires several actions related to the microwave, it actually turns out to be quite complicated. The player is forced to think about every step that needs to be done, when in real life it is something that she may do without thinking. Restoration of behavior here turns an everyday action into a performative act, almost akin to how Brecht's acting in Epic Theatre, where the actor is aware of the artificiality of every movement involved in the performance (121-2).

Thus, choosing the right level of abstraction in a game is a fundamental tool to establish how the player can restore the behavior in the game. Higher levels of abstraction are more likely to dictate a specific behavior to be restored, because there is less room for behaviors that deviate from the one expected from the game. Therefore, restricting the number of objects to interact with and the possible actions associated with them facilitates generating the story, simply because there is not much else left for the player to do. On the other hand, lower levels of abstraction give more leeway to the player to explore the world and experiment, but also make it more likely for the player to perform in ways that will not allow her to advance in the game. Nevertheless, adventure games also have other ways to help the player restore the proper behavior in the game, such as giving clues in the descriptions of the game, creating expectations using a

specific genre, limiting the spaces that the player has access to, or the time that a certain action can be made. These devices will be covered in more detail in the ensuing chapters.

### **5.5.2. Levels of Abstraction in Game Mechanics**

The historical trend towards higher levels of abstraction in adventure games has also been reflected in the interface of adventure games, to address questions of usability and playability (Fernández Vara, “Shaping Player Experience”). Adventure games have gone from having to type the commands in natural language, which allowed for a relatively large range of actions, to clicking on an object to perform an action, which the game itself selects depending on the context. This shift in the interface has had a direct impact on the mechanics of the game.

Higher levels of abstraction in the mechanics makes the interaction less error-prone—the program does not have to return an error message whenever she performs an unsuccessful attempt to do something. Immediate feedback and reduction of error messages are qualities that improve the usability of computer programs (Shneiderman 494). This is a way to address the problem that posed by Lebling, Blank, and Anderson when they referred to how to implement the simulation efficiently—the player should be solving the problem in the game rather than the program (52). However, being more efficient in the interaction can also get in the way of experimenting and exploring the world. Trial and error still constitutes a basic strategy to learn the rules of the simulation. Increasing the usability in this case can also affect the range of actions, preventing the player from exploratory actions that help her learn more about the fictional world.

Adventure games can have a relatively large list of verbs: text adventure games could have relatively powerful parsers, which would accept a good range of verbs. The

original Zork, for example, would recognize “71 distinct ‘actions’” (Lebling, Blank, Anderson 53). Graphic adventures, such as *King’s Quest V*, would present the player with five verbs (go to, look at, pick up, talk, use (inventory item), examine (inventory)), plus the save / load commands and audio/visual settings of the game. Current point-and-click games, such as *Sam & Max: Season One*, have done away with the verb list. The player clicks on an object, and the game chooses for the appropriate action—clicking on a character is equivalent to choosing “talk to”; clicking on a closed door will be interpreted as trying to open it.

Thus, when the mechanics select higher levels of abstraction, the player does not have to choose a verb at all; rather, the program chooses which action to apply depending on the what the player is trying to interact with. This design strategy resorts to “context mechanics,” as Sicart describes them, “mechanics that are triggered depending on the context of the player presence in the game world.” Therefore, every object only has one action associated with it, simplifying both the design and the interaction with the game, as was the case of *Myst*. Higher levels of abstraction did away with verb-hunting, but it also got in the way of exploring and experimenting with the world.

## **5.6.Conclusion**

A simulation is the implementation of the fictional world of a game as a rule system. Adventure games are simulations, precisely because they present a simulated fictional world the player interact with. However, the shift towards higher levels of abstraction, both in relation to the simulation and the mechanics, has lead some authors like Aarseth to affirm that they are not (“Genre Trouble”). The simulation concretizes the fictional world, so that the player can inhabit and interact with it.

The remaining chapters deal with the specific aspects of the simulation that bridge the game with the story.



## CHAPTER 6

### THE GAMENESS OF ADVENTURE GAMES

#### 6.1. Puzzles, Stories and Performance

Adventure games may be “story-game” hybrids, but little attention has been paid to how they are actual games and how they are designed. It is difficult to understand adventure games in terms of rules, as we may understand a board game like *Monopoly*, a racing game like *Mario Kart: Double Dash*, or a FPS like *Half-Life 2*. Adventure games are based on puzzle-solving, although their puzzles are different from how *Tetris* is a puzzle game, for example.

The puzzles in adventure games are usually integrated in the fictional world, and are part of the simulation. Since the simulation is where the elements of story and game meet, solving the puzzles also means advancing the story—as we saw in the previous chapter, that is the expected restoration of behavior. The consecutive states of affairs achieved by solving puzzles are the changes of state that constitute the story events. Thus, adventure games relate to literary works that also include game-like elements, such as detective stories (Suits) or riddles (Montfort).

The difference between reading a detective novel and playing an adventure game lies in the fact that the game is performative—it depends on the participation of the player to take place. Reaching the solution of the puzzle means restoring the behavior expected to produce the performance. The player has to perform the steps as designed in order to complete the game. As we stated in Chapter 2, this can also be pleasurable because it implies “synchronizing” with a preset pattern. Its performativity is also

different from theatre—the player is not given an explicit script or a set of instructions, but rather has to find out what the steps of the performance are.

This chapter will deal with the “gameness” of adventure games, and how it establishes the performance—what the nature of its game design is, how it is based on puzzle-solving, how puzzles relate to stories, and what types of puzzles we can find in adventure games.

## **6.2. Adventure Games as Games of Progression**

The dichotomy between games of progression and emergence was already introduced in Chapter 4. According to Jesper Juul, adventure games are the archetypical example of games of progression, which “directly set up each consecutive challenge in a game” (*Half-Real* 67). Juul contrasts progression with games of emergence, which “set up challenges indirectly because the rules of the games interact” (67). This basic distinction is based on how the challenges to the player are structured. Progression and emergence seem to be the opposite ends of a spectrum, but as Juul argues most games incorporate elements of both progression and emergence (82).

Juul defines games of emergence as the historically dominant form, where a few rules can generate complex dynamics when set into motion, as in traditional games (e.g. Parcheesi, Poker); they are the essential model for the MDA framework proposed by Hunicke, LeBlanc, and Zubek. Games of progression, on the other hand, are the more modern model: “Progression games are a historically new game form, where the game designer explicitly determines the possible ways in which the game can progress” (Juul *Half-Real* 56). This new form seems to be associated with games that try to incorporate stories into the game; Juul also mentions that these types of games tend to have

“cinematic or storytelling ambitions” (73). Juul’s remark seems to echo Costikyan’s observation in “I Have No Words and I Must Design” that games and stories traditionally have been different things, and only in recent decades have they been brought together. The attempt to integrate stories and games has been more pronounced in videogames; the properties of digital environments allow media convergence, i.e. multiple media (e.g. moving images, written word, sound recordings) to be co-presented using the same technology. The incorporation of stories into games is relatively new, since it refers to the last 30 years or so, although it is not exclusive to videogames—*Dungeons and Dragons*, on the one hand, and riddles on the other (Montfort, *Twisty Little Passages*), are older forms that probably laid down the foundations of games of progression (as in, for example, dungeon crawling). It is no coincidence that both *Dungeons and Dragons* and riddles are two of the cornerstones of adventure games as well.

Adventure games can indicate progression by using scores. Many text adventure games, and some graphic adventure games gave points to the player whenever a puzzle was solved. They also displayed a total score, (e.g. “Score: 0 of 342” in *Gabriel Knight: Sins of the Fathers*) to indicate how many points it was possible to score in that game. The player can usually finish the game without scoring all the points; this is a way to prompt players of the explorer type (Bartle, “Players Who Suit MUDs”) to replay the game to try to get a complete score. The score is an extradiegetic convention borrowed from other videogames, as a way to indicate progression. However, scores to indicate progression have been dropped by more recent adventure games, where it is usually assumed that the player may only play the game once.

The defining trait of adventure games as games of progression is that each

challenge appears consecutively (Juul, *Half-Real* 67), thus “progression” alludes to “linearity” without using the word. Linearity is constantly invoked as a criticism to adventure games, by Aarseth (“Genre Trouble”), as well as videogame journalists, fans, and game designers (e.g. Costikyan, “Games, Storytelling, and Breaking the String”). However, the term is seldom defined, especially in the case of adventure games. Juul in *Half-Real* devotes far more space to discuss different aspects of games of emergence, sidelining games of progression and their relation to stories. Adventure games present the opportunity to expand the definition of games of progression, and also specify what the term “linearity” refers to. By elaborating on that definition, we can also draw further differences and parallels between games of progression and emergence.

#### **6.2.1. Adventure Games are Multisequential**

‘Non-linear’ is an ill-defined concept in videogames. It is usually coupled with “narrative” or “storytelling”, and it is a phrase constantly invoked to refer to the properties of narrative in new media, ranging from hypertext, to Interactive Fiction. Murray prefers to use the terms “unisequential” and “multisequential” in reference to digital stories (Murray, “Building Coherent Plots in Interactive Fiction” 18-9). Unisequential means “consecutive” along a single path; the sequence of events always take place in the same order. Some adventure games, such as *Syberia*, seem to definitely follow the “beads on a string” structure that Costikyan talks about (“Games, Storytelling and Breaking the String”).

Multisequential, on the other hand, refers to the multiplicity of ways in which the same events can be generated. For our purposes, we are making a distinction between “multisequential” and “multiform” (Murray, *Hamlet on the Holodeck* 30-8), which refers

to the possibility of generating different events in a simulation. The story generated by interacting with the simulation of adventure games is usually the same, and although the events may take place in different order (multisequential), it is always the same. Some adventure games may include different events depending on some specific decisions of the player, but frequently they are anecdotal, or only affect how the game ends (e.g. *Indigo Prophecy*).

As Rollings and Adams argue (457), story-driven adventure games tend to have a clustered structure, where the action is divided into chapters. Each chapter has a series of interdependent challenges integrated in the simulation (the puzzles), and the player cannot move on until all of them have been solved. Thus, the player may move from chapter to chapter in a sequential fashion, but the problems posed within each chapter may not have to be solved in a particular order. The solution can be multisequential; there may be some that do not have to be tackled at all. This does not seem to be very different from other games, where you cannot advance to the next level until you have completed all the challenges of the previous one: First-Person Shooters like *Half-Life 2*, mission-based flight simulators like *The Secret Weapons of the Luftwaffe*, or fighting games like *Street Fighter II* have modes with a similar structure. These games use segmentation of gameplay (Zagal, Fernández Vara, and Mateas), which is a design device to manage where the players goes and what she can or cannot do. Segmentation of gameplay therofre affects how restoration of behavior takes place. Games thus can have traits both of emergence and progression; these examples show how segmentation of gameplay helps introduces progression in games of emergence.

As we saw in Chapter 2, the puzzles in adventure games are concatenated, i.e. the

solution to a puzzle may require solving several other puzzles first; once that solution is reached, other puzzles will probably show up. The causal relationships between puzzles is what establishes that concatenation. A great example of concatenated puzzles is the spitting contest in *Monkey Island 2: LeChuck's Revenge*. The contest is not a really a competition, but a set of puzzles. In order to win, the player character, Guybrush Threepwood, has to purchase two different drinks and mix them to thicken his spit. The drink only gets him so far—he needs to do more if he wants to win. In the contest area itself, there are several stones marking how far other contestants got with their spitting. Guybrush has to create a distraction for everyone to look away by blowing a ship horn, which makes a crazy pirate shoot a cannon because he thinks the mail ship has arrived. While the contest crowd is looking away, Guybrush has to move the stones back, so it is easier for him to reach further than other contestants. Finally, in order to win the contest, Guybrush has to time his spit with the wind blowing in his favor.

All these actions form a sequence, although buying the drinks and buying the ship's horn can be done in any order. It is the completion of all of these conditions (drinking the mix, shifting the stones, spitting at the right time) that eventually allows the player to win the spitting contest. It is not going from point A to point B, it is solving the puzzle to fulfill three programmatic “if statements” in order to change the game state.

There are games where the player may not have to solve all of the puzzles in the game. For example, in *Gabriel Knight: Sins of the Fathers*, there are some things that the player can do, but are not necessary to complete the game. The player can talk the owner of a snake to dance in order to pick up the dead skin it has left behind. These extra puzzles give an extra depth to the simulation, they are the type of actions that the player

can try out. It is the leeway the player is given from restoring the behavior—exploration of actions is one of the defining traits of the adventure games genre, as we saw.

What makes the progression of adventure games be perceived as unisequential are not the connections between different puzzles, but *how* the puzzles are solved. Adventure games usually have a single solution, which is typical of puzzles where there is only one state of affairs that can be considered the answer. Thus, adventure games usually do not present multiform stories.

Adventure game puzzles are rarely the result of a strategy or grinding, but rather they are the result of a moment of illumination, of insight—either you get it right or not. Juul explains this feature as “there are more ways to fail than to succeed” (*Half-Real* 73), a statement that, in general terms, is also true of most other genres. It is usually easier to lose than to win a game, or else there would be no challenge. What is different is that in adventure games there is only one way to get to that solution, and that it involves insight thinking (Danesi). Insight means that the person solving the puzzle learns something new, inferred from the puzzle (Danesi). This is how riddles are solved—a situation is set up, and the solution is usually reached after an “a-ha” moment, rather than a series of activities of sub-challenges. Montfort points out the relationship between Interactive Fiction and riddles in *Twisty Little Passages*—they are challenges and verbal literary works, which engage the reader/interactor to solve them (37-63). This dissertation proposes to extend that relationship to all adventure games, because most puzzles in adventure games have a similar structure to riddles—one single solution, achieved after a moment of insight.

### **6.3.Gameplay Rules: Puzzle Solving**

Adventure games combine stories and games in a segmented form. Since progressing in the game means advancing in the story, both the gameplay and the story have to be broken down. Segmentation of gameplay makes it easier to combine the sequences of events in the story with the consecutive states of affairs that are achieved through solving each puzzle. Thus, performing in the game involves completing all these segments—each segment can be considered an individual behavior that must be restored.

The main form of gameplay in adventure games is puzzle-solving; the segmented nature of adventure games can also accommodate mini-games as well. Puzzles differ in many ways from other types of games, though they are a form particularly related to stories already—some puzzles tell a short story that poses the challenge that must be solved.

#### **6.3.1. Defining “Puzzle”**

In general terms, a puzzle is a type of challenge, where the opponent is not another player, but a specific problem that needs a solution or an explanation. Rollings and Adams refer to puzzles as “mental challenges” rather than challenges that require physical skill (Rollings and Adams 229). They also qualify them as “static,” given that there is no active agent against whom the player is competing.

Marcel Danesi points out the relationship between puzzles and mysteries: “[T]hey generate a feeling of suspense that calls out for relief” (Danesi 2). The appeal of the puzzle, according to Danesi, is precisely that it is a problem that requires a solution; they are a gap waiting to be filled. The impulse to fill that gap, to solve that problem, is what he calls “the puzzle instinct,” and what attracts players to puzzles. The allure of puzzles is



their ambiguous nature, because at they conceal the answer at the same time they demand it, spurring the player to measure their wit against whoever created the puzzle (Hovanec 10). This ambiguity is at the core of the gameplay of adventure games—the player is invited to play both to solve the puzzle and learn more about the story. The solutions to the puzzles cannot be too easy, or else there is no challenge, but if the solution is too concealed it will not be found. Achieving the solution is satisfying because it closes that gap, solving the mystery.

The mystery of the puzzle is a game both for its author and the person trying to solve it (Suits)—rather than being at odds, the designer is challenging the player. A puzzle is not a tug-of-war—if the player is stumped by the puzzle, it may be a failure on the part of the designer if there was not enough information to solve it. Even when the player is given the solution, the logic of the puzzle should make sense to the player, otherwise the player will feel cheated.

Puzzles usually have just one right answer, although there may be several ways to find it. This is true of riddles, which look for one specific answer, or jigsaws, where there is only one right way to fit all the pieces. In the case of jigsaws, as Kim notes, the solution is not the result of a moment of insight, but rather of a consecutive series of insights, where the player figures out the position of the pieces in relation to others one by one. Some puzzles may have several ways to reach their solution, e.g. in the Eight Queens Puzzle, there are many ways to arrange eight queens in a chess board so that none of them is in a position to capture another. The state of affairs required in this puzzle can be fulfilled by several solutions, but it is still a single condition.

This discussion allows us to come up with a basic definition of puzzle: A puzzle

is a challenge where there is no active opponent, but rather it is a problem that needs a solution. The solution entails logical thinking, rather than physical skills, and it is the result of insight thinking. Puzzles usually have a single solution, even if it may be possible to obtain it in more than one way.

#### **6.4. The Relationship Between Puzzles and Games**

Scott Kim defines puzzles as follows: “A puzzle is fun and has a right answer” (35). This definition highlights that puzzles are different than solving problems in everyday life—we solve puzzles for fun, not because we must. Kim highlights “right answer” as a defining feature of puzzles to make a distinction between puzzles, games, and toys. According to Kim, games do not have an answer, and toys do not have a goal. This distinction seems consistent with what Danesi calls puzzle games, which seem to include puzzles based on pre-existing games (e.g. Eight Queens puzzle) and arrangement games such as the Towers of Hanoi, or the family of tile-matching videogames. Kim and Danesi both point out that puzzles are of a different nature from games, although neither of them clearly states what the similarities and differences are.

Puzzles and games are related—both take place apart from everyday life; playing a game or solving a puzzle is something that we do for its own sake. The answer of a puzzle may correspond with the win state of a game, and we saw above we can make games that consist of solving different variations of a puzzle, such as *Bejewelled*. A game like *Tetris* is based on the pieces of a puzzle, Pentominoes, which have to be arranged to make lines. The game has the added challenge of having to move the piece as it falls fast to its place—skill and reflexes are then involved, making *Tetris* the perfect hybrid between game and puzzle.

In order to explore the relationship between games and puzzles, we can compare their rules as their basic building blocks. A game has a set of rules, which set up the constraints to the actions the player can perform, and a goal, which establishes the win state. A game may also have objects, which have rules attached to them, such as chess pieces or the ball in dodge ball. On the other hand, a puzzle is a problem that needs a solution. The problem poses a question that must be solved, usually setting up the constraints within which the solution must be reached. A puzzle can involve objects too (e.g. a jigsaw), though the physical component can be drawing or writing (section puzzles, crosswords), or it can be a statement (e.g. a riddle).

Puzzles are therefore a very specific type of game, where the rules correspond to the conditions in which the problem must be solved. The main difference consists of the type of goal: the goal must be an answer, a solution, the element that helps close a gap. By having a different type of goal, puzzles also have different dynamics. One does not “win” a puzzle, but rather solves it, because a puzzle is not a competition against anybody else. Solving a puzzle means reaching a specific state of affairs that helps complete a mystery, something that is missing. The solution is usually the result of insight (Danesi), so that finding the solution means learning something that we did not know before.

An individual puzzle is difficult to understand in terms of the MDA framework, since insight is not much of a process, and the solution does not suppose a change of game state. However, a series of concatenated puzzles, which is what we find in adventure games, do create dynamics—each puzzle solved provokes a change of state in the simulation, thus constituting an event in the story. Adventure game puzzles set a

specific gameplay pace, usually more relaxed than in action games—the player can usually take her time to solve the puzzle.<sup>23</sup> Players can also come up with different strategies to solve the puzzles, such as examining every object and talking to all the characters, hoarding every object available, or mapping the space. These dynamics also create a different experience, one that encourages the player to explore and allows the player to use trial-and-error as a way to figure out how the world works.

#### **6.4.1. Finding the Solution: Insight Thinking**

Marcel Danesi anchors his discussions on puzzles on the importance of *insight thinking* in order to solve a puzzle. According to him, “[i]nsight thinking can be defined as the ability to see with the mind’s eye the inner nature of some specific thing” (28). The solution to the problem posed provides novel knowledge; that solution is in the “inner nature” of the problem, so that the person who solves the puzzle learns something new. The solution is also surprising, witty, out of the ordinary, which increases the sense of novelty—it is what distinguishes solving a riddle from solving a mathematical problem. Achieving insight is also pleasurable, and the novelty of the knowledge is also an achievement.

Most importantly, insight is the result of the previous knowledge of the person achieving it, and it must be the result of certain logical processes. As Danesi states: “Insight thinking does not emerge fortuitously or haphazardly. It comes about only after the observation and contemplation of recurring patterns” (28). That is, the insight is the product of what the player already knows, and the connections between ideas and

---

<sup>23</sup> Timed puzzles, which will be dealt with in Chapter 10, are an exception.

concepts that she has already made. Solving a puzzle helps establish new connections between ideas.

Robert Sternberg describes three different ways in which insight thinking uses reflective memory: selective encoding, selective comparison, and selective combination (Sternberg 80-1; also quoted in Danesi 28).

Selective encoding means making apparently irrelevant information relevant. This is a typical tactic in riddles and minute mysteries, also in puzzles where you have to find the hidden image. The puzzle maker hides the solution to the player, and the player has to discover it. Hidden object games are an example of a genre that thrives in selective encoding, since the player has to make out what an object is and where it is hidden in a picture.

This riddle from Lewis Carroll is an example of the use of selective encoding:

Dreaming of apples on a wall,  
And dreaming often, dear,  
I dreamed that, if I counted all  
—How many would appear?<sup>24</sup> (*Lewis Carroll's Games and Puzzles* 4)

Selective comparison is the use of analogies and metaphors in order to draw a non-obvious relationship between two pieces of information. Again, this is a basic tactic of riddles, which in essence are metaphors or similes. These literary tropes represent an object or concept (the tenor) in terms of another object or concept (the vehicle) that has some proximity or relation of similarity to it (metonymy), such as shape, color, or one being part of another (synecdoche) (see Baldick). The puzzle maker has to come up with

---

<sup>24</sup> The solution is evident when you read the second line as “And dreaming of ten, dear.”

the analogy, and the riddle usually consists of the vehicle—the player has to guess the tenor from it.

Hovanec gives this example, from the monk Claret in 14<sup>th</sup> century:

A vessel have I  
That is round like a pear,  
Moist in the middle  
Surrounded with hair;  
And often it happens  
That water flows there.<sup>25</sup> (15)

Selective combination happens when different pieces of information are merged in order to form a novel one. This type of insight applies to many types of different puzzles: jigsaws (the individual pieces form a whole), mathematical puzzles such as magic squares (grids where a series of numbers have to be distributed according to a specific condition), and acrostics (the final solution is made up by the first letters of words, which are also the solution of a riddle). In adventure games, this is probably the most common type of puzzle. Building machines, cooking, or using a key in a lock are all examples of selective combination puzzles.

Sternberg's descriptions of the process by which we reach insight thinking are somewhat limited to account for any type of puzzle. However, these three types of insight seem to describe very well the types of puzzles that we find in adventure games. They can also be useful categories in order to come up with strategies to design puzzles—the designer can start from the information the player has to find out, and then think of what

---

<sup>25</sup> Solution: the eye.

kind of insight is needed to solve the puzzle. Being aware of these strategies gives the designer basic ways in which to encourage the restoration of behavior—the player will have to perform a certain activity, which will lead to the appropriate change of game state.

Most of the solutions to puzzles in adventure games can be reached by following one of these three processes. The pieces of information usually manifest as different objects, and the player must perform specific actions using those objects, which is one of the distinctive features of the genre. Let us see how each of these three processes are exemplified in *The Secret of Monkey Island*.

Selective encoding: The player character, Guybrush Threepwood, has to buy an expensive map to find the treasure of Mêlée Island. However, when he buys it, he finds out it is a diagram for dancing steps instead. He confesses to the player he believes he has been swindled; it is up to the player to figure out that the diagram is the real map. The dance steps are actually a set of instructions for navigating the forest and getting to where the treasure is buried. The diagram seems like a piece of irrelevant information, since the player character believes it is not the map he needs. The player has to realize that it actually provides relevant information to find the treasure, because it tells her what exit to take on each screen (up, down, left, or right).

Selective comparison: Guybrush needs a job, and there is a circus in town which needs someone to test their cannon. He needs a helmet in order to become the test human bullet. There are no helmets on the island, but a cooking pot is close enough—especially one where, as the player character notes, “someone has cooked a headcheese.” The employers think that the pot is good enough as a security measure and hire Guybrush.

Selective combination: One of the puzzles requires the player character to cook a magical potion, for which he has the recipe. However, Guybrush is on a ship in the middle of the ocean and he does not have all the proper ingredients, so he must find substitutions: for example, breath mints can work as leaves of mint, and red wine as two pints of monkey blood.

These three types of insight are not mutually exclusive; very often the types of insight needed to solve a puzzle overlap and or compliment one another. In the case of the potion, for instance, the player must figure out the relationship of similarity between the substitute ingredients. These different categories of insight help us understand how puzzles in adventure games work and how they may be solved. These types of insight also help us evaluate puzzles in a game—if the solution of the puzzle cannot be achieved by using any of these three types of insight, then the puzzle is either poorly designed, or not even a puzzle.

Providing insight is fundamental in adventure games, because they help the player see the world in a new light and learn new things (Danesi). As we saw, the fictional world is implemented in the simulation. Puzzle-solving helps the player learn more about the fictional world through the interaction with the simulation, both about the fictional world and its rules. In the examples from *The Secret of Monkey Island* above, the potion puzzle teaches the player that voodoo magic works in this world. It also tells the player about the pre-history of the game, since the recipe for the concoction is found in a diary, which narrates how the crew of a ship got to Monkey Island after cooking it.



#### 6.4.1.1. Designing the Puzzle

The cornerstone of solving a puzzle is finding a piece of missing information that brings about a new state of affairs, based on previous knowledge of the player. The nature of that knowledge and how the relationship is established is difficult to define. Knowing how everyday things work, such as locks or switches, or about how certain activities take place, such as buying something, is the basis for most puzzle design. If the knowledge needed is more concrete, the fictional world can appeal to certain domain knowledge, such as cooking, topography, or shoe-mending. A fictional world belonging to a specific genre can also invoke specific domain knowledge—the crime scene in Infocom's *Deadline* invokes detective work, such as finding evidence or cross-questioning. If the knowledge is only relevant to the fictional world, and the player cannot be expected to know it beforehand, then it must be provided by the game itself throughout the game. The recipe for the potion in *Monkey Island*, for example, is something that the player cannot know from everyday life, therefore the game provides the recipe, and finding out the equivalent for each ingredient becomes another set of puzzles.

By learning more about the world, the player can create expectations about how to solve later puzzles. This knowledge, as well as the consecutive insights achieved, prepares the player to perform correctly as she progresses in the game. Since the player has to discover the script of the game (i.e. the behavior to be restored), she must have the information that allows her to complete that script.

Players must have all the information necessary to solve the puzzle, either from their own world knowledge, or from what they have learned about the fictional world.

When the connection between pieces of information only happens in the designer's mind, and does not have any evident connections with the domain knowledge, the player will not be able to solve the puzzle other than by trial and error. This is what is popularly called a “designer’s puzzle” (Bates 128). In this case, the designer may not be aware that the player does not have all the information needed, or that the connection is only evident to him. This type of design usually makes the player feel frustrated with the game; it is usually decried as forcing one to “read the designer’s mind.”

The abundance of “designer puzzles” has been blamed as one of the reasons for the supposed death of adventure games. Erik Wolpaw describes one of the puzzles in *Gabriel Knight 3: Blood of the Sacred* to explain how puzzles in adventure games got to be ridiculously complicated and follow dubious logic.

Gabriel Knight must disguise himself as a man called Mosley [sic]<sup>26</sup> [...]. In order to construct the costume, Gabriel Knight must manufacture a fake moustache. Utilizing the style of logic adventure game creators share with morons, Knight must do this even though **Moseley [sic] does not have a moustache.** [...]<sup>27</sup>

[...] [Y]ou might think that you could yank some hair from one of the many places it grows out of your own body and attach it to your lip with the masking tape in your inventory. But obviously, Ms. Jensen [the game designer] felt that an insane puzzle deserved a genuinely deranged solution. In order to manufacture the moustache, you must attach the masking tape to a hole at the base of a toolshed then chase a cat through the hole. In the real world, [...] this would result in a piece of masking tape with a few cat hairs stuck to it, or a cat running around with tape on its back. Apparently, in Jane Jensen's exciting, imaginative world of books, masking tape is some kind of powerful neodymium supermagnet for cat hair.

[...] At the end of this puzzle, you have to affix the improbable cat hair moustache to your lip with maple syrup!

---

<sup>26</sup> The correct spelling of the name is Franklin Mosely, Wolpaw misspells it both times.

<sup>27</sup> Emphasis in the original.

Wolpaw's rant, in which he concludes that adventure games "committed suicide," is a great summary of one of the recurring problems of adventure game puzzles. The problem is not the overt and unnecessary complication of the puzzle (the spitting contest puzzle in *Monkey Island 2* was far more complicated), but the disconnect between how things work in the real world and in the game. This is not a problem in itself, since fictional worlds can work differently from the real world, as in the sci-fi universes of *Planetfall* or *Space Quest*. The problem of the fake moustache puzzle is that it neither follows an everyday logic, nor is it properly set up in the game—as Wolpaw mentions, the person Gabriel Knight has to pass for does not have a moustache to begin with. It does not fit the tone of the game either—an off-the wall puzzle like this may fit a humorous game, such *Sam & Max Hit The Road*. But the *Gabriel Knight* series is supposed to be a "serious" detective saga with supernatural elements.

As we saw in Chapter 4, restoring the behavior in adventure games is enjoyable, because the player gets to re-enact the events of the story; in puzzle-solving, insight is the key to being able to fulfill that restoration. It is not a matter of being immersed in the game, but a matter of eliciting pleasure from performing what seems correct in the context given. If it has not been properly set up in the simulation, the behavior will not seem proper, and synchronization of the player with the story will be broken.

The lack of set up in the simulation turns gameplay into sheer trial-and-error—trying to combine each item with every other item, until something happens. Although trial-and-error is a form of experimentation which the player can use as a tool to learn how the world works, forcing the player to apply combinatorics instead of logic spoils the appeal of adventure games. It is even worse, because when something happens and the

puzzle is solved, the player fails to see the connection between the problem posed (produce a makeshift moustache) and the steps following the solution. The “puzzle instinct” that Danesi was talking about is not satisfied, since the mystery does not seem to be solved.

Providing insight about how the simulation works is the main challenge of designing puzzles, particularly in the realm of adventure games. The natural ambiguity of the puzzle, which conceals its solution and yet entices the player to solve it, creates an interesting crux for game design. If the solution to the puzzle is too obvious, it is not compelling, since it is not a challenge anymore. On the other hand, if it is too difficult to figure out, or if the solution is not really logical, the puzzle becomes too frustrating.

Puzzles that encourage lateral thinking are probably the most effective, since they are not straightforward, but they still require following a logical, if unusual, path. The real challenge is to design puzzles that, by providing insight about the fictional world and by extension of the simulation, also make the player feel smart because she has discovered something new, and she was able to restore the proper behavior.

#### 6.4.1.2. Types of Puzzle Logic

Sternberg’s classification of processes to reach insight helps us understand the ways of thinking necessary in order to solve a puzzle. Based on Sternberg’s processes, there are certain types of thinking that are more commonly appealed to in the design of adventure games.

##### *6.4.1.2.1 World use of an object*

The player has to resort to her knowledge of the world (either everyday life or the

fictional world) in order to figure out the puzzle. *And Then There Were None*, a game based on Agatha Christie's eponymous novel, thrives on this kind of puzzle, since it takes place in a storyworld that is not fantastic—the player has to find batteries in order to fix a flashlight, or find the key to open a drawer.

#### *6.4.1.2.2 Lateral thinking*

Lateral thinking puzzles are based on selective comparison, where the player has to find a solution that may not be self-evident. These puzzles have a logical solution, even if it is a surprising one. The pot-as-helmet puzzle in *Monkey Island* is an example of this type of logical thinking.

#### *6.4.1.2.3 Designer Puzzles*

Designer puzzles have been addressed in section 4.4.1.1.

#### *6.4.1.2.4 Trial and Error*

Trial-and-error is a basic strategy for players to learn more about the simulation. The basic strategy is “What happens if I do this?” and seeing the result. This is the basic way to explore the world, although too often it becomes the strategy players follow when they do not know what to do in the game. When the player fails to achieve insight because she cannot make the connection, or does not have enough information, or cannot read the designer's mind, solving the puzzle becomes a matter of combinatorics, where the player tries all the verbs available (or that she can think of) with all the objects, and tries to combine all objects together. This was considered acceptable by designers like Roberta Williams at the beginning of her career:

Don't be afraid of trying ideas. Try everything, say everything. [...] Sometimes the most outlandish things will solve a puzzle, sometimes only the most logical. That is why you need to try everything. Most adventure games have provisions for saving and storing your games along the way. Save your game often. Especially right before you try to solve a puzzle or fight a monster. [...] Sometimes a sneaky author will throw in something that you can carry around, that does absolutely nothing, but basically, objects have a use. Usually objects are used only once or only at one place, but not always. Weapons seem to have a higher percentage of being used more than once.

Roberta Williams here is talking about earlier games where the influence of *Dungeons and Dragons* was still patent—there was combat and the player character could die, forcing the player to start from a previous moment in the game that she had saved. This approach to design encourages the exploration of actions, attempting to solve a problem in many different ways. The problem with enforcing trial and error is that, although it prompts exploration of the actions, it may take away the key pleasure of puzzle-solving, which is achieving the “a-ha!” moment. It also enforces repeated failure as part of the restoration of behavior, rather than letting the player fail on her own. Again, if the player does not see why a certain action or set of actions is the solution to a puzzle, the feeling of “closing the gap” is not achieved, and it is not really a puzzle.

Trial-and-error in certain games usually forces the player character to die in order to obtain more information. Graham Nelson complains about this practice, because it takes the player out of the game, requiring the player to make use of what she has learned in “other lives” and apply it to the game (*The Craft of Adventure*), thus the information needed to solve the puzzle is extradiegetic. However, this has been the source of interesting effects in Interactive Fiction pieces such as *Aisle* and *Shrapnel*. In *Aisle*, the player only has a single turn to perform an action, and then the game is over. The player can then replay the game many times in order to obtain all the endings, where some of the

actions come from the information given after the one command allowed. In *Shrapnel*, the knowledge of past lives is integrated (and satirized) in the story itself. The player character dies and leaves a corpse behind, which its following reincarnation can look at and examine.

## **6.5.Puzzles in Adventure Games**

The simulated world in adventure games is where the puzzles are embedded. The objects and characters in the simulation are both the *existents* of the story and the elements that the player has to interact with in order to solve the puzzle. Thus adventure game puzzles are related to the story-puzzle, the riddle, or the minute mystery: the story is posing a problem that the player has to solve. For example, in *King's Quest V*, the player character, King Graham, sees how his castle and his family are spirited away by an evil wizard, so he sets out in order to find who the wizard is and how to get his family and his castle back.

In order to solve this initial problem, which becomes the goal of the game, there will be other sub-goals that need to be addressed first. This is very clear in *The Secret of Monkey Island*, where the player character wants to be a pirate (it is the first line he utters in the game). In order to do so, he must show proficiency in sword fighting, finding treasures, and thieving. These initial situations have to be sorted out by solving a series of puzzles—the final goal recedes from the player as the new sub-goals are introduced.

### **6.5.1. Performativity and Puzzles**

The puzzles of the game are integrated in the simulation; thus by solving the puzzles, the player is also performing in the simulation. That performance, as we saw,

consists of completing a pre-established set of steps, each step being determined by the puzzles. As we saw, this establishes a different pacing from other games.

Puzzle-solving also gives way to a specific method of performance, perhaps more detached than in other games. The player must take a critical stance towards the world, becoming aware of every step needed to solve a puzzle—opening a door is a trivial everyday action; by turning it into a puzzle, it highlights the action itself. Lower levels of abstraction, as we saw in Chapter 5, can break down trivial actions into a chain of behaviors to be restored, thus defamiliarizing the action. Games with gestural interfaces can recreate the alienation effect by highlighting the artificiality of an everyday gesture, as Brecht described in his principles of Epic Theatre (121-129).

Puzzle-solving in adventure games can thus create a sense of de-familiarization, even with everyday activities. Adventure games can undermine the concept of immersion in digital environments, since performing in them means adopting a critical distance and having to be aware of every action in the world and its effectiveness. In Chapter 8, we will see how the player character is another device that can undermine the immersion in the game.

Puzzles in adventure games also create patterns, in a similar way to dance or music. These patterns are not only a way to create a rhythm for the game, they also help the player figure out the rules of the simulation and apply them. *Myst* is an excellent example: each Age employs certain predominant actions in order to solve the puzzles. In the Channelwood Age, every mechanism on the island works by running the water current through it, so every puzzle in that area involves routing the water current in the pipes to each mechanism. In the Selenitic Age, sounds are the key to solve the main



puzzles. The player has to orient a radio tower to specific locations in the island where certain sounds can be heard. Later, the player has to ride a vehicle through a labyrinth; a sound plays whenever the vehicle reaches a crossroads, so the correct path is in the direction where the same sound plays again.

The fictional set up of the puzzles is also at the service of restoring the behavior—as we will see in the following chapters, the space and the characters also help the player become a performer in the simulation.

### **6.5.2. The Relationship between Puzzles and Stories**

We referred to Nelson’s definition of adventure games in Chapter 2, “a crossword at war with a narrative” (“The Craft of Adventure”). Adventure games are like a crossword, in that they are a series of puzzles (riddles, in the case of crosswords) which the player has to solve. As in crosswords, the solutions to the puzzles are interdependent and interrelated, so that solving one puzzle becomes part of solving the rest of the puzzles.

The story of the game is a puzzle in itself—it is fragmented and scattered in pieces (e.g. object descriptions, pictures, events). The player has to put the story together, finding out the causal relationship between the different fragments, figuring out the patterns of the story and the puzzle. Playing an adventure game thus means reconstructing its story.

Puzzles also hold strong ties with storytelling. To begin with, many puzzles already are a literary form, as Montfort argues when he compares riddles with Interactive Fiction (*Twisty Little Passages*). Some mathematical puzzles and logical problems are also presented as a short story. Raymond Smullyan’s collection of logic puzzles *The Lady*

*or the Tiger? And Other Logic Puzzles* frames each group of puzzles within a specific setting, such as an insane asylum, where the reader has to find out who is sane and who is not based on their statements and using logical deduction. Smullyan links one puzzle to the next, as well as their solutions, to make up a narrative. Stories can also make puzzles part of their narrative, such as Lewis Carroll in *Alice's Adventures in Wonderland* and *Through the Looking Glass*.

Puzzles are very much related to mystery, as Danesi points out; mystery and detective stories are particular literary genres that take advantage of our “puzzle instinct” to makes us keep on reading—Bernard Suits tackled detective stories as a game, both for the writer and the reader. Edgar Allan Poe’s short story *The Golden Bug* incorporated puzzles in its narrative; he is also credited with inventing the modern detective story with the adventures of August Dupin, such as “The Murders in the Rue Morgue,” “The Mystery of Marie Roget,” and “The Purloined Letter.”<sup>28</sup>

A clear evidence of the relationship between the detective genre and puzzles is how mystery stories also have to set up the mystery, and provide enough information for the reader to figure out the mystery on her own, even before the solution is revealed. If vital information was withheld from the reader during the narrative, it will be impossible to figure the mystery out. As in puzzles, even if the reader had not figured out who the murderer was before the end, it must be possible for the reader to go back and realize that she was given all the information, the narrative must still make sense after the mystery is revealed. If not, the author will be thought to have cheated to the reader (if we follow on Suits concept that mystery stories are a game both for writer and the reader).

---

<sup>28</sup> See Introduction by Dawn B. Soba in Poe.

The structure of detective stories, and more specifically of the *whodunit* genre, also fits very well with videogames. Todorov characterizes whodunits as a dual genre—there is the story of the crime, and the story of the investigation. “The first story, that of the crime, ends before the second begins. [...] The characters of this second story, do not act, they learn.” (McQuillan 122). Jenkins also points out the dual story structure for videogames (“Narrative Architecture”), which also corresponds to the two basic types of stories we identified in relation to games—the pre-history of the game, which is the history of the fictional world, and the player’s history, which is the series of events that unfold during gameplay. In games that follow a detective story model, the main puzzle is uncovering and assembling the pre-history of the world, from *Deadline* to *Broken Sword: The Shadow of the Templars*, the *Gabriel Knight* series, or even *Myst* to a certain extent.

The detective story model seems to be the easiest to relate to games, though other types of stories can also result in interesting games, such as the coming-of-age in *The Secret of Monkey Island*, or fairy tales in the *King’s Quest* series. It seems easier to integrate the goal of the game in a mystery story, because solving the mystery usually poses a clear goal.

Puzzles also integrate more easily with stories because they have a single solution, which guarantees that the player will have to reach a single, specific state of affairs. This state of affairs is where story and game can meet, so that solving a puzzle is a change of state in the simulation, marking a milestone in the progress of the game and of the story.

Puzzles are therefore a way for the game design to control what the player has to do, guiding the player to the appropriate behavior to restore. If a puzzle has more than

one solution, it can also multiply the potential story events, branching the story according to the different outcomes. Multilinear stories present a series of design problems, since the story is produced by the designer/writer of the game, rather than being produced programmatically (Crawford). Using branching in a discretionary way can be manageable from the point of view of design, but trying to make it the foundation of the game makes it not only very difficult to produce, but also more prone to inconsistencies in the story as gameplay unfolds.

The story can limit the possible types of puzzles to a few, which usually involve either obtaining information or an object, or having something happen in the simulation in order to bring about a specific state of affairs. Thus, it is less common to have certain types of puzzles, such as mathematical puzzles, jigsaws, anagrams, or even riddles themselves. These other types of puzzles usually stand on their own: although solving them helps advance the story, they do not entail the existents of the story. When they appear, they do so almost as “games within the game,” what we will call mini-games (as we will see below). As we saw in Chapter 5, the relationship between the existents of the story and the elements needed to solve the puzzle is what marks the difference between adventure games and puzzle games with a story setting, such as *The 7<sup>th</sup> Guest* or *Professor Layton and the Curious Village*. In these games, the insight needed to solve the puzzle does not relate to the fictional world; the new information we are given is a reward for solving a puzzle, but not a discovery on the part of the player.

### **6.5.3. Types of Puzzles**

The puzzles in adventure games are restricted to a few specific types, as we said above, because of their relationship with the story. Game designer Bob Bates lists

nineteen types of puzzles in games, all of them inextricably tied to the fictional world (121-127). His understanding of puzzle design is heavily influenced by adventure games, and his categories are divided between do's and don't's. His types of puzzles are more of a laundry list than an actual systematic classification of puzzles. The prescriptive nature of the list, as well as the absence of a system, makes this classification unwieldy for this dissertation.

The following game puzzles types are based on the type of action that needs to be performed, according to the type of interaction that needs to take place (navigation, object manipulation, exploration). All these actions are relevant in the fictional world, and help advance the story, as Bates argues (119-20). The main categories according to the actions required are: navigation puzzles, state of affairs puzzles, language puzzles and mini-games.

#### 6.5.3.1. Navigation Puzzles

Exploring the world is one of the first things that the player of an adventure game has to do, finding where the player character can or cannot go, and where the player can gain access to new areas. In the case of text adventures, the first challenge was actually mapping the space—since the space was all described in text, the player had to draw a map in order to visualize where to go, and where each object was. This could be a challenge in itself, since the spaces could be rather labyrinthine.

Navigation challenges usually require selective combination in order to be solved. The player must figure out the relationship between locations, in order to put together a new piece of information: the map of the game. We will tackle navigational puzzles in the following chapter, which is devoted to spaces and their design.

#### 6.5.3.2. State of Affairs

The most basic type of puzzle is achieving a specific state of affairs in the simulated world by interacting with the objects and characters in it. The goal is to bring about a change in the game state to be able to continue playing and advance the story.

Several of these puzzles may make up a larger puzzle. For example, the goal of first set of puzzles in *Space Quest I: The Sarien Encounter* is to escape a spaceship which is going to self-destruct in fifteen minutes. In order to do so, Roger Wilco, the player character, has to find a keycard that will open the door to the pod bay, where an escape pod is docked. Before getting on the pod, Roger has to don a spacesuit and helmet; in order to escape, he has to fasten the seatbelt in the pod and press the right button. All this must be done before the spaceship explodes, always avoiding the Sarien guards who have assaulted the ship and killed the rest of its crew. In order to escape the ship, a specific state of affairs is needed: spacesuit on, seatbelt on, and pressing the right button.

The way that the world is affected is usually through the use of transitive verbs (i.e. verbs that need an object, such as “talk to,” “use,” or “open”) constitutes the basic interaction with the simulation that defines the genre. However, it must be noted that text adventures, and early graphic adventures (including the original version of *Space Quest*), include intransitive verbs in their vocabulary, such as “wait,” “swim,” or “jump.” Intransitive verbs have mostly been dropped from contemporary adventure games, so that all the interaction of the game revolves around using objects or interacting with non-player characters.

Some of the actions that the player may have to perform in order to solve a state-of-affairs puzzle are: use the right object with the right verb, use the right object with

another object/character, manipulating the environment puzzles, or do something within a specific time limit.

#### 6.5.3.2.1 *Using the right object with the right verb*

This type of puzzle is particularly common in games with a large list of available verbs, such as text adventures or menu-driven games like *Eric the Unready*. This is the basic type of puzzle in adventure games—choose the right action within the specific situation concept. It can be a simple action like opening a drawer in order to find an object, or doing something more specific, such as turning sheep green in order to camouflage them (*Loom*).

As we saw in Chapter 5, later adventure games have done away with verbs completely, so that using the right verb is not a problem. The incorporation of context mechanics (Sicart) means that the action is selected by the rules and performed after the player clicks on a specific object and character. For example, clicking on a door will make the player go through it (e.g. *Sam & Max Season One*, *Ceville*). *Indigo Prophecy / Fahrenheit* presents a relevant variation in context mechanics: the player is given a series of options depending on the context. For example, in the very first chapter of the game, the player can hide away a corpse in a bathroom stall. Only after the corpse has been hidden will the actions with relation to a specific object be available—a mop, which the player can pick up. After picking it up, the player can choose to put the mop back where it was, or mop the blood trail left by the corpse.

In games that incorporate gestural interfaces, this type of puzzle becomes performing the right gesture at the right time. In some cases, the player is given the cue of what to do. Again, some of the interactions in *Indigo Prophecy / Fahrenheit* are gestural:

dragging the corpse above requires the player to move the analog controller / press the left-right arrow keys rhythmically and quickly, as a way to mimic the effort of dragging something heavy. In other cases, the player has to find out the gesture that has to be made—in one the puzzles in *Hotel Dusk: Room 215*, the player has to complete a jigsaw, moving the pieces with the stylus. Then the player has to find a message on the reverse side. In order to read it, the player must physically close the Nintendo DS console, so that the jigsaw flips from one screen to the other. The player has to figure out this gesture, rather than following explicit cues.

#### 6.5.3.2.2 *Using the right object with another object/character*

This type of puzzle is based on selective combination; in this case, rather than ideas, the solution consists of using two things together. The most basic example of this type of puzzle is the lock-and-key puzzle—find a key that will give you access to a new area or object. As we saw, in *Space Quest: The Sarien Encounter*, Roger Wilco needed a keycard in order to access the pod bay. The lock and the key do not have to be literal; a lock-and-key type of puzzle can also consist of giving an object to a character in order to obtain something else. In *The Secret of Monkey Island*, a troll guards a bridge that gives the player access to the east part of Mêlée Island. The troll wants something that will attract attention, but have no real importance, in order to let the player character pass. The object that he wants turns out to be a fish, more specifically a red herring. Here we see how different types of insight are combined in order to solve a puzzle, in this case selective comparison (understanding the clue that the troll gives the player character) that should lead the player to understand the metaphor literally, and then selective combination (giving the herring to the troll opens up a path in the island).



#### 6.5.3.2.3 *Manipulating the Environment*

Environmental Puzzles are similar to using an object with another object, but in this case what is important is where the object is placed in the environment. This type of object is a variation on jigsaw puzzles, such as reconstructing a map from its pieces. *Indiana Jones and the Fate of Atlantis* presents several puzzles that consist of arranging objects, such as replacing pieces of a machine to fix it, or aligning three discs on an axis. Environmental puzzles are common in *And Then There Were None*, where the player has to repair a radio in a basement using different pieces from other electric devices, and find materials to fix an apple presser.

#### 6.5.3.2.4 *Timed puzzles*

One extra aspect that digital environments bring to puzzles in adventure games is that they may be timed, so that the player has to achieve the specific state of affairs within a time limit. The beginning of *Space Quest: The Sarien Encounter* gives several examples of how the time limit refers to the puzzle. The time limit can be applied to a specific action: Sarien guards are patrolling the spaceship, so that periodically Roger can hear their steps. That gives Roger just a few seconds to clear the area where he is, normally by getting on an elevator that will take him to another floor. There can also be a time limit to a set of puzzles: the player has to perform all the actions necessary to escape before the end of fifteen minutes.

Timed puzzles change the dynamics of the game, as mentioned above. The player has limited time to think about what to do, or may not be aware of the time limit, so these puzzles often involve trial-and-error. The first time the Sarien guards show up, for example, they will probably hunt down Roger, just because it takes a couple of tries to

learn how to escape from them. We will analyze this type of puzzle in more depth in Chapter 10, where we will tackle how time is regulated in adventure games and how it affects gameplay.

State-of-affairs puzzles are the most common type of puzzles; players, when they are stuck in a game, try combining all objects and using all verbs with them. They are the most likely to fall into the “designer puzzle” trap and become a trial-and-error puzzle, since the designer is aware that the player may eventually find the solution just by experimenting with the simulation.

#### 6.5.3.3. Language Puzzles

Language puzzles are more traditional, closer to riddles, and are inherent to adventure games where the input is textual, although they can be found in graphic adventures too. Language puzzles usually require selective comparison in order to be solved, since they thrive on metaphors and similes.

##### *6.5.3.3.1 Find the right word*

The player has to find the exact term, or the exact phrase that will result in a successful action. This type of puzzle, rather than designed, tends to be the result of a poor word parser, or a lack of anticipation on the part of the designer about what kinds of actions the player may try.

One of the most famous examples is the magic word “xyzzzy,” which made its first appearance in the original *Colossal Cave / Adventure*. This magic word is found scribbled in a note in one of the rooms; whenever it is invoked while in that room, it takes the player character to the Building in the game, and vice versa. “Xyzzzy” has become an

Easter Egg in many other Interactive Fiction pieces,<sup>29</sup> including *Zork*, where the invocation is answered with “A hollow voice says ‘Fool.’”

At times, the right word requires the player to go outside the vocabulary that refers to the world itself. In *Zork*, for example, there is a Cyclops who the player can chase away by typing “Odysseus,” in reference to the episode in the *Odyssey* where Odysseus/Ulysses tricked the Cyclops Polyphemus and blinded him.

#### 6.5.3.3.2 Riddles

Apart from the potential of Interactive Fiction as a new type of riddle, riddles themselves can also be puzzles in adventure games. They require selective comparison, since they are a type of metaphor, as we explained above. However, riddles are not exclusive of text adventure games—the “red herring” puzzle from *The Secret of Monkey Island* already mentioned is in essence also a riddle.

#### 6.5.3.4. Exploration Puzzles

This type of puzzle does not require any type of insight, just exploring the world exhaustively; in a way, they are not really puzzles. The player has to examine every nook and cranny of the game: speak with everybody, find all the objects in the world and examine them (to find other objects or clues given in the description of the object).

Although not technically a puzzle, exploration provides the player with clues to solve a puzzle of another type. Bates notes that the distance between the information given and the puzzle where it is needed is an indicator of the difficulty of the puzzle

---

<sup>29</sup> See <http://webhome.idirect.com/~dswxyz/sol/xyzyy.html> for a list of games that use XYZZY as a command.

(131), since the player has to remember a piece of information that was given earlier in the game. These exploration puzzles are a cue for selective encoding—the player has to find the information, which at first will probably seem irrelevant, but will be useful later in the game.

A rich simulation offers a deep world to explore, and provides consistent and extensive information about how it works. This was the key, for example, of *Myst*, where the nameless player character was transported to a parallel universe. The player had to figure out what machines were for and how they worked in order to solve the puzzles. The clues were written in the books in the main library, as well as in different notes scattered through the island. The player has to explore the island, as well as every age, to learn the logic of this parallel universe and to find out where to apply the information given in the books.

Exploration puzzles can take one of the following forms: finding information, triggers, or hunting the pixel.

#### *6.5.3.4.1 Finding information*

The player can find new information by examining objects, which prompts a written/oral description of the object, or by talking to the characters in the world. It is the basic type of puzzle in detective games such as *Gabriel Knight: Sins of the Fathers*, which bases most of its gameplay on finding information. Gabriel learns about crime scenes and relevant events by through conversations with other characters.

#### *6.5.3.4.2 Triggers*

At times, by exploring the world the player makes something happen, often

unexpectedly. Again, this is not technically a puzzle, since it does not require insight thinking, but it ends up affecting the world. An example of this can be when talking to a non-player character makes that character do something necessary for the player to progress. Again, in *Gabriel Knight: Sins of the Fathers*, if Gabriel asks the Voodooine about her snake, she will demonstrate how she dances with it. This is something that is done through conversation, rather than exchanging objects, and requires the player to exhaust all the dialogue options.

#### 6.5.3.4.3 *Hunt the pixel*

This type of puzzle was brought about by graphics and, more specifically, the point-and-click interface. This type of pseudo puzzle requires the player to explore the screen with the mouse, pixel by pixel, in order to find a specific object, usually hidden or very small. It is an example of selective encoding, although it does not really ask the player to see the world in a different way. It requires the player to sweep the screen with the cursor exhaustively and carefully. In *Indiana Jones and the Last Crusade*, Indiana Jones has to pull a bookcase, to find a small piece of adhesive tape behind it. The adhesive tape is just a couple of pixels, and the player only finds out that it is a hot spot by moving the mouse very carefully.

This type of puzzle has created its own genre, as we saw in Chapter 2, the “hidden object” games. Since it requires patience rather than insight, again, it’s not really a puzzle; the exploration does not really give more information about how the world works, but rather about how many objects can be found in the location.

#### 6.5.3.5. Mini-games

A mini-game is a game whose rules only apply to the section of the simulation where it takes place, and whose goal is separate from the rest of the game. In Chapter 5 we already introduced how mini-games are hypodiegetic, using Montfort's term (*Twisty Little Passages* 29-31). Mini-games usually happen at a different level in the simulation; they are games-within-the-game. Their outcome may affect the simulation, although this is not always the case: *Sam & Max Hit The Road* includes four games which the player can play without affecting the events in the game: a car-themed version of battleship, a highway-surfing game, a coloring book and a paper doll dress-up game.

Mini-games also usually change the level of abstraction in the simulation, often identified by showing a different interface, and by giving the player a specific goal. In the case of graphic adventures, they also expand the number of verbs of the game. For example, in the highway-surfing minigame in *Sam & Max Hit the Road*, "jump" becomes an available verb, the only one during that period.

There are two general types of mini-games: traditional puzzles and games-inside-the-game.

##### *6.5.3.5.1 Traditional puzzles*

These include logic puzzles, jigsaws, math puzzles, or word puzzles. For example, in *And Then There Were None*, one of the puzzles consists of re-arranging a set of drawers, which are identified with a letter each. A poster in the nearby room reads "Rule Britannia!" which gives a clue of how to arrange the drawers. Once they are in the right orders, the letters in the drawer read "Rule the waves", which is part of the second verse of the song. This puzzle requires the player to be familiar with this particular hymn, or

else it will be more difficult to solve.

#### 6.5.3.5.2 *Game-inside-a-game*

Other times, we have games inside the game, rather than puzzles, with their own rules and goals. A good example of this is found in *Sherlock Holmes and the Serrated Scalpel*: the eponymous hero must win a game of darts in order to obtain information from a character in a pub. When the game of darts starts, the interface and the controls change—the player must press the space bar in time, in order to aim where to throw the dart. The particularity of this game is that, once the player wins, she can go back to the pub and play any time, which contributes to the consistency of the world—why should Sherlock Holmes not be able to take a break and play darts whenever he wants?

Another example of how mini-games can be integrated in the narrative is *Indiana Jones and the Fate of Atlantis*. In order to provide the player with higher replay value, the game had three different modes: the Wits Path (the gameplay is based almost exclusively on puzzle-solving), the Fists Path (the game includes action mini-games, such as fighting, and easier puzzles), or the Team Path (which is also heavy on puzzles, but this time the player has to control two player characters, Indiana Jones and Sophia Hapgood, the heroine of the story). The different modes thus incorporated different mini-games, from a car chase, to piloting a submarine, providing the game with varied gameplay that borrowed from different genres. The inclusion of mini-games usually means that they are relatively simple games, usually based on a game the player is already familiar with; the player is not supposed to play them for a long time.

An extreme example of games within games is found in *Day of the Tentacle*, the sequel to *Maniac Mansion*. The player can find a computer which runs the full version of

the previous game. It is not a mini-game any more, but a full game within another game, establishing a new diegesis within the diegesis.

## **6.6.Conclusion**

Puzzles are the main form of gameplay in adventure games, and mark how the player performs in the game. In this chapter, we have identified different types of insight thinking required to solve a puzzle, different types of puzzle logic, and different types of puzzles according to the types of action required to solve them. These different typologies are as inclusive as possible, although there may be still some types of puzzles that may not have been covered.

Puzzles are the key gameplay element to integrate gameplay and story. Restoration of behavior hinges on puzzle-solving—the solution of the puzzles makes the story advance. Although very much related to other types of puzzles, adventure game puzzles are a unique form because they are integrated in a simulated world, their concatenation being the element that allows them to integrate as a series of events in a story.

We are not done with puzzles yet, though—in the following chapters we will deal at more length with specific types of puzzles that relate to the existents of the story. Navigating the virtual space, as we saw, can be the source of different types of challenges; time regulation can also be a device in puzzle design. Conversation, as a way of obtaining information as well as having characters do something, is the last design element that can also become a puzzle. These puzzles also have multiple sub-categories, and therefore will be dealt with in relation with each specific existent.



## CHAPTER 7

### PERFORMANCE SPACE

#### 7.1. The Qualities of Videogame Space

Spatiality is a fundamental quality of digital environments (Murray, *Hamlet on the Holodeck*), and by extension of videogames. Videogames take place in a digital space created through code, and represented textually (as in the case of text adventure games) or graphically.<sup>30</sup> Murray points out that, although other media can represent space, such as paintings or novels, digital environments offer the possibility of *navigating* those represented spaces. We can refine that observation by specifying that they do not only allow navigation, but also manipulation of the space—in the case of videogames, the player not only navigates but interacts with the entities and objects contained in the space. Manipulation within the space also shapes the type of interaction that takes place in it.

As we saw in Chapter 2, space is also one of the fundamental elements of performance (Schechner, *Performance Theory*), since the performance needs a location in which to take place. In the case of games, this performative space is the “magic circle” (Huizinga; Salen and Zimmerman, *Rules of Play*), the space that separates the play activity from the real world, and within which the rules of the game are in effect. In videogames, the space is virtual, i.e. it is a simulated physical space which can be inhabited. Thus the constructed virtual space is also a space of performance.

---

<sup>30</sup> A few games have represented space aurally, such as Dark Room Sex Game. See list at <http://www.audiogames.net/>

In this chapter, we will determine the qualities that define space, and how they apply to videogames. Then we will analyze the properties that differentiate adventure game spaces from other games, and how by performing in the simulated space, the player discovers and re-enacts the story of the game.

## **7.2. Defining Space**

The definition of space has to go beyond a physical concept—after all, our focus is virtual spaces, which exist as a digital fabrication. In the real world, the space is defined by the actions of the people in it, whether the space is a natural landscape (a valley, a beach) or architectural (a building, a street). There are different features which will help us understand how the action defines the space: practice defines the space, the activities in the space make it polyvalent and transformative, and the performance defines the space itself.

### **7.2.1. Space is Defined by Practice**

Michel de Certeau differentiates between space and place. A place (*lieu* in French) is a configuration of elements, a set of positions and interrelationships between those elements that denotes stability. “Place” thus denotes immobility, a set of elements that do not change, as if frozen in time. On the other hand, space (*espace* in French) is the result of the movements that take place in it, of simultaneous processes of orientation, situation, temporalization, and conflict. As Certeau puts it, “Space is a practiced place” (Certeau 117). Videogame space is one of continuous movement and reconfiguration between elements, thanks to the participation of the player. Thus we should talk about videogame *spaces*, rather than videogame *places*.

### **7.2.2. Space is Polyvalent and Transformative**

By being defined by practice, space is defined by what people do in it, so that different activities can make the same space transform depending on what takes place in it. Space is thus polyvalent (Certeau 125), referring to the different relationships (e.g. social, historical) that can define the participation within it. This polyvalence between the elements of the space can also give way to different interpretations by the observer (Lynch 6), so that the space transforms depending on what happens in it and who is looking at it.

The polyvalence of the space and its transformative qualities can be observed in everyday life. Different activities can take place simultaneously in one space, or one activity can predominate for a given period of time, making the space versatile. For example, a city square can be a crossroads, a meeting place, or a hub of commercial activity. It can also transiently become a concert venue or the host to a demonstration or a celebration. The transformative qualities of the space are what allow games to take place almost anywhere. As we saw in Chapter 4, a school playground can turn into a makeshift football field, or a garden becomes a playground for hide-and-seek as soon as players start the game.

### **7.2.3. Space is Performative**

Participation in the space can also be understood as performance, especially if that participation is within the “magic circle” that separates the activity from everyday life. A performative act defines the space and the way that is perceived. Theatre, again, provides us with the clearest example of how this happens. Peter Brook starts *The Empty Space* with the following quote:

I can take any empty space and call it a bare stage. A man walks across this empty space whilst someone else is watching him, and this is all that is needed for an act of theatre to be engaged. Yet when we talk about theatre this is not quite what we mean. Red curtains, spotlights, blank verse, laughter, darkness, these are confusedly superimposed in a messy image covered by one all-purpose word.” (9)

Brook strips theatre to its very essentials—to have a stage, you only need an empty space, and it will be the action of an actor, along with someone watching that action, and it becomes the most basic theatrical act. As we saw in Chapter 4, the audience is an essential part of the performance (see also Schechner, *Performance Theory* 14-19; Pavis). In this process, the space acquires consecutive new meanings—first from space to stage, and then whatever relationship the actor establishes with the stage. Depending on how the actor walks, the stage may become a street, a gym, or a waiting room.

Elizabethan theatre is the epitome of how the stage is defined by action. Shakespeare’s plays were originally performed on a stage without backdrops, which was the same for all plays. The actors in their costumes, their voice and movement, what they said, along with a few sparse props, transformed space in the eyes of the audience. The opening speech of *King Henry V* (I.i.) describes how that process takes place—it is the “imaginary forces” of the audience that bring the fields of France, castles and palaces to the stage, and move the action “here and there, jumping o’er times.” The stage is the signifier, and the actions on it are what changes its signified. The space is thus defined by semiosis, by how we understand it, before we take into account any of its physical properties.

Thus there are different ways in which the space can be defined by the actions

that take place in it. Videogames, as a participatory medium, need the practice to exist; it is that practice that will define the space. The space of videogames, unlike theater, is not empty—rather than being transformed by the performance, it has to give the cues to the specific behavior that is to be restored. Practice is regulated by the design of the game, and designing the game is inextricable from designing its space.

### **7.3.Space in Videogames**

How do the above qualities of space (defined by practice, polyvalent, transformative, performative) apply to videogames? If in theatre action defines space, in videogames the space creates the action. The videogame space is designed to support the actions that are supposed to take place in it. The design of the space has to be both at the service of the interaction in it, as well as of the fictional world of the game (Squire and Jenkins).

Action designs the space in order to create a specific experience; different actions will encourage different spatial configurations. For example, a level of a First-Person Shooter (FPS) is different from a 3D platformer, even if both games simulate three-dimensional spaces. While FPSs are designed as a hide-and-seek space, with corridors and with props that allow taking cover, platform games must encourage jumping and climbing by having platforms at different levels, as a way to avoid enemies or reach items. Thus, devising the space is a fundamental aspect of game design, which is thus a discipline related to architectural design or urban planning (see Squire and Jenkins; Jenkins “Narrative Architecture; Borries, Walz and Bottger).

On the other hand, polyvalence is not a common quality of videogame space, since it is designed for specific types of navigation and interaction. Versatility can be a

very valuable quality, albeit a difficult one to achieve. In *Super Mario 64*, for example, the player traverses each level in different ways, depending on the specific goal given. *Castlevania: Symphony of the Night* takes place in a castle that turns upside-down in the middle of the game, so that the player has to traverse it again inverted.

The versatility of the space also depends on the level of abstraction in the simulation. The richer the simulation is the wider the range of actions that can be performed by the player, and thus the more different types of restoration of behavior which can take place. This not only refers to the actions that are possible in the space, but also to uses of the space that were not anticipated by the designers. Lower levels of abstraction not only encourage emergent behavior because of the more nuanced interaction, but also create more versatile spaces. A great example of this is *Grand Theft Auto: Vice City*. The player can choose to drive through Vice City following traffic rules, or perform car stunts, or be a cab driver and earn money by taking people to their destination, or drive an ambulance—all are behaviors encouraged and supported by the space. Depending on the goal the player is pursuing and what behavior she chooses to restore, the space will be experienced differently—a player who wants to do stunts will look for potential ramps, whereas someone playing as a cab driver will slow down while looking for potential passengers.

The videogame space allows for multilayered access (Nitsche, *Video Game Space* 35-6), which means that there can be multiple inputs during gameplay: from pressing buttons simultaneously, to mouse/analog stick and even voice input. Online FPSs are a textbook example of this. Multilayered access also allows experiencing the space in different ways, since interactions with the simulation happen at various levels.

### 7.3.1. Affordances of the Designed Space

Videogame spaces have a range of affordances, which can be harnessed as part of the design of the game. The basic affordances of the videogame space are navigation, participation, and being part of the story setting. Each of these main affordances has a subset of design elements, which refer to the specific actions that can be performed in the space.

#### 7.3.1.1. Navigation

Navigation is one of the basics of digital environments and, by extension, of videogames, as we saw above. This affordance, however, presupposes that there is always an avatar that the player is guiding through the space. There are videogames where the space is difficult to understand in terms of navigation—again, the puzzle game genre (*Tetris*, *Puzzle Bobble*), as well as digital implementations of card games (e.g. *Solitaire*, *Hearts*) put this general definition into question.

##### *7.3.1.1.1 Movement*

Even if it is difficult to account for *Tetris* in navigational terms, it is nevertheless true that we can affirm that there is movement within the space. Movement is what transforms a place into a space, as Certeau said, because that is how the relationships between the different entities are established (117). Thus, the pieces in *Tetris* fall from the top, the player can rotate them and move them before they become lodged in the bottom.

The rules determine what type of movement can take place. How the game entities can move within the space is defined by the cardinality of the gameworld

(Fernández Vara, Zagal and Mateas), which in our terms refers to the general rules of the game. For example, in *Pac-Man*, the ghosts and Pac-Man can move up and down and left and right, whereas in *Super Mario 64* all the entities can move within the three-dimensional space. There can be a discrepancy between the cardinality of the gameworld (which is regulated by the general rules of the game) and the cardinality of gameplay, i.e. how the player is allowed to move the entities she controls (which is regulated by the game mechanics). This discrepancy can become an obstacle that the player has to work around. *Galaxian* is a clear example of this—enemy ships move side-to-side most of the time, but they can plunge towards the player and loop around, whereas the player can only move the ship left and right to avoid or shoot enemies.

Designing how the player has to move in the space is basic to how the player experiences the world, but it is not the only device in relation to space at the service of game design, as we will see in the sections below.

#### 7.3.1.1.2 Exploration

Navigation at the service of exploration is one affordance that is generally not encouraged as often as it should be. There is a definite pleasure in discovering every nook and cranny of a game, especially if it is adequately rewarded. *Castlevania: Symphony of the Night* hides many secrets, e.g. walls that can be torn down to access hidden chambers, or attics that are difficult to access. By finding those secret sections of the game, the player is rewarded with power-ups and other items that can make her gameplay easier.

The encouragement of exploration is more common in games with rich fictional worlds. As we saw, the fictional world of a game does not have to be fully fleshed out before becoming a simulation, since the simulation will abstract that fictional world in the



implementation. However, the more complete the fictional world is, the more material will be available to encourage exploration. Through exploration, the player can discover more about the pre-history of the game. Going back to *Castlevania: Symphony of the Night*, the player can learn more about who the player is in many sections of the game. A ferryman in the underground refers to him as a “prince”; one of the hidden chambers causes the player character to have a nightmare where he meets his mother, who was burnt at the stake. We will deal with how the space integrates the story in the sections below.

#### 7.3.1.2. Participation

The space of videogames is also participatory (Murray, *Hamlet on the Holodeck*)—it not only encourages specific actions, but it can also be affected by the interactions of the player. The spaces of *Pac-Man*, for example, cannot be physically modified by the player; however, there are other cases where the player can operate with the objects in the space to change it. A basic way to affect the space is by gaining access to other areas, by opening a door, for example. In *Half-Life 2*, the player can move crates and then jump on them in order to reach a place high up. The interaction does not have to limit itself to the physical reconfiguration of the space, it can also change the relationships between the different entities contained in the space—the crates that were an obstacle turn into a helpful item.

##### *7.3.1.2.1 Contest*

The space of the game can primarily be one of contest and struggle—after all, competition is one of the basic types of games (Huizinga, Caillois). Squire and Jenkins

refer to game design as the art of contested spaces (Squire and Jenkins). Two basic patterns of interaction in relation to space are competing for who dominates the space (i.e. fighting *for* the space), as in *Civilization III*, or struggling to get from point A to point B (fighting *in* the space) as in *Super Mario 64*. In the case of *Civilization III*, we see how expanding one's civilization transforms the space, not only because the territory becomes larger, but also because it becomes more difficult to manage and to protect from outside attacks. *Civilization III* echoes Certeau's allusion to shifting boundaries telling the story of a place (126-7)—the player's history thus becomes the history of the space as well.

#### 7.3.1.2.2 Puzzle-Solving

The space can also be reconfigured in order to reach a specific game state. Rather than fighting for or in the space, the contest can turn into a fight *with* the space itself. Reconfigurative interaction usually refers to solving puzzles, which we dealt with in the previous chapter. The example of *Half-Life 2* above shows how reconfiguring the space allows for a specific action. Another example can be found in *The Legend of Zelda: Phantom Hourglass*, where the player has to place four statues on four different switches in order to lower a grate and access a new room. Since the predominant gameplay of adventure games is puzzle-solving, we will expand on this below.

#### 7.3.1.3. Story Setting

The fictional world provides the setting of the story, as we saw in Chapter 5. The two types of stories in relation to the fictional world (pre-history and player's history) are also the stories of the space: what has happened in the space before and what happens as

the player interacts with the simulation. In the case of the detective story model, introduced in the previous chapter, the story of the player revolves around uncovering the story of the space.

Thus the design of the space can also become creating a story for the game. In this line of discourse, Henry Jenkins affirms that game designers are less storytellers and more “narrative architects” (Jenkins, “Narrative Architecture” 121). Michael Nitsche extends this argument: “[...] [G]ame spaces evoke narratives because the player is making sense of them in order to engage with them. Through a comprehension of signs and interaction with them, the player generates new meaning” (*Video Game Spaces* 3).

Again, semiosis is an inherent part of the creation of the space, as well of putting together the events that have taken place in the space—the story itself is a puzzle. By turning the videogame space into the setting of the story, the player can make better sense of the simulation, and of the puzzles and the game entities contained in it. The space itself is one of the richest aspects where game design and story meet. The following sections discuss the different design elements that contribute to the integration of the story in the space: navigation, indexical storytelling, and enacted stories

#### *7.3.1.3.1 Navigation*

The basic activity that brings together game design and story is navigation, the movement of the player traversing the space. This is particularly true of games where there is a player-controlled entity that the player guides through the space, as is the case of adventure games, CRPGs, or FPSs. A game such as *Civilization III* or *The Sims* is difficult to understand in terms of navigation because the player does not control a single entity, and can have multiple viewpoints on the space and the events that happen in it. In

these games, there may be gameplay-relevant events that happen off-screen while the player is busy doing something else.

The movement in the space is linear, from here to there; stories, also linear in nature, can therefore take advantage of navigation to trace a path that will make the story unfold. By designing where the player can go, game design can control the player's story in the game and organize her experience.

The relationship between navigation and story is not new—as Murray points out, navigational storytelling in digital environments relates to the journey story (Murray, *Hamlet on the Holodeck* 139). We should add that there may be different types of journey stories, such as the travelogue and the quest. The travelogue chronicles the events of a specific journey, where the goal is the journey itself. The quest is the journey towards a achieving a goal, which usually relates to transcendence, the transformation of the hero who is assigned the quest. The quest is another story model that has been appropriated by videogames, as a model to provide the player with a goal in relation to a story. Role-playing games in particular base their gameplay on the quest model, as described by Joseph Campbell or Vladimir Propp.<sup>31</sup>

The difference between travelogue and quest is relevant to game design, since we can draw parallels between them and the concepts of games of progression and emergence. A travelogue recounts a voyage, whose goal is exploring new territories, without knowing what will be found. Games of emergence can thrive on a similar eagerness to experiment, to venture by poking the system of the game. On the other hand, the goal of the quest is completing a specific task, so that the story is about how the

---

<sup>31</sup> See Howard for a detailed discussion on CRPGs following the Quest model.

different obstacles that got in the way were overcome. The quest also entails following a series of stages or steps, as described by Propp and Campbell—the quest therefore marks the progression of the traveler towards the goal. Story-driven games can thus combine progression and emergence by wavering between the narrative genres of the travelogue and the quest.

#### 7.3.1.3.2 *Indexical Storytelling*

The space itself can tell a story, without words or without having the player follow a specific path. This has been called “environmental storytelling” (Carson), based on strategies borrowed from theme park design (see also Pearce, “Narrative Environments”). Environmental storytelling is a broad concept, which usually refers to a catalog of different design strategies.

It is necessary to introduce a more specific notion, *indexical storytelling*, to talk about how a space can contain a story independent of navigation and appealing to the interpretation of the player of the space. While navigation constructs the story of the player in a specific order, the space itself can also “tell” its pre-history. Player interaction happens in a specific point of time, after a series of previous events have taken place and left a trace in the space. For example, the frontiers between countries are the result of historical struggle and negotiation, of peoples fighting, advancing, migrating, returning (Certeau 126)—this is the type of storytelling that we referred to with *Civilization III* above.

The events that took place in the space define it historically, so they become indices in the Peircean sense. The indices are the consequences of an event, which point to something that happened or is going on, inviting the player to reconstruct the events. A

classical example of an indexical sign is smoke coming from a forest, which denotes that there is a fire going on. A type of index that is more akin to videogames is a footprint. Footprints indicate that someone has been in a specific location, where the person went, whether the person was standing, running or jumping, and how long ago. In short, indexical storytelling encourages detective work—it is how the player can find out the story of the game.

*Bioshock* is an excellent example of indexical storytelling in the space. At the beginning of the game, the player arrives in Rapture, a lavish underwater city. But something must have gone wrong—the city is populated by insane, disfigured individuals, and the whole place looks in disrepair. The game uses indexical storytelling to show how there has been struggle and protests—there are placards littering the floor. People tried to leave, as evidenced by the abandoned suitcases scattered around what looks like some sort of commuter station. However, there was no way out—the panel indicating departures displays “cancelled” in every line. Without a word, the player can learn quite a bit of information just by looking around, and associating the different objects in the space.

A concept related to indexical storytelling is Nitsche’s “evocative narrative elements.” These elements are the building blocks that structure the player’s experience and facilitate the understanding of the simulation as the player “reads” them (*Video Game Spaces* 37). The contribution of these elements to the game resides in how the player figures out the relationship between them. “Creating these connections, players can form narratives that refer to the gameworld” (44). What is relevant here is the idea that the player is making sense of the world, and connecting ideas. Indexical storytelling requires

the player to invest some time considering what she sees in order to make sense of it, since the game does not stop and tell the story to the player—it is integrated within gameplay. This type of storytelling constitutes another type of puzzle in itself, since the player has to put the pieces of the story together.

#### *7.3.1.3.3 Enacted Stories*

We have covered the story of the player (navigation) and the story of the space (indexical storytelling); there is another design device where both the story of player and space meet: enacted stories. Henry Jenkins has coined the term “micronarratives” to refer to short narrative units in the game (“Narrative Architecture” 125). These units are elements within a larger body of work that are supposed to produce an emotional impact. Jenkins’ concept seems to echo Certeau’s “micro-stories”, which refers to the “family stories” or “life stories” that have taken place in a space, within a larger socio-cultural context (125). Certeau’s concept illuminates why Jenkins highlights the importance of emotional impact—micro-stories relate to specific individuals, to characters that the player can identify with, rather than with larger socio-historical events.

Micronarratives can be cut-scenes, but they can also be “a series of pre-programmed actions” (Jenkins, “Narrative Architecture” 125). Jenkins thus seems to imply that these micro-narratives are not interactive, the player has no control over them as they take place. During cut-scenes, for example, the player has no agency, just sees the events that affect the world and her character. The fact that they are pre-programmed also seems to indicate that these are not the result of the system dynamics, but they are triggered when specific conditions are met. These stories are “enacted,” as Jenkins puts it, but the player seems to be more of a spectator than a participant in that enactment.

As we argued in Chapter 5, micronarratives do not necessarily interrupt gameplay, although still the player may not be a participant in them. They are part of the performance, mostly because the computer is the one performing those pre-scripted micro-narratives. *Half-Life 2* is a typical example—as the player navigates and interacts with the space, events are triggered and the player has to react to them. Micronarratives are milestones in the linear navigation of the game, allowing the game makers to control what parts of the story the player will experience, when, and where. *Bioshock* presents an alternative example, where the player finds recordings scattered through the city. Every recording seems to be a personal diary, which tells of a different event (e.g. a New Year's party, an experiment gone awry) that took place where the recording was found. The player does not have to listen to the recordings; if she does, she can listen to them as she plays the game. All these micronarratives tell different fragments of the story of the city of Rapture and help the player understand better what has happened. They also require the player to listen to them and to make the connection between the recording and the space of the game. Thus micronarratives can also be a type of evocative narrative element, especially when they are connected to what happened in the space.

So far we have been dealing with the space in games in general. Let us discuss how the concepts introduced above apply to adventure games.

#### **7.4.Space in Adventure Games**

Space is a basic building block of adventure games. A room is the basic spatial unit of adventure games, and moving between rooms is the most elementary type of interaction. The space of adventure games is designed to restore a very specific behavior—the one that will re-enact the embedded story of the game. Thus space in



adventure games is also mostly at the service of the story—adventure games not only create narrative spaces, but use design devices that are almost exclusive of the genre, as we will see below.

Nelson (“The Craft of Adventure”) argues that the term “room” is probably inherited from Crowther’s *Adventure*, since it is a commonly used term in caving to refer to “any discrete space, no matter the shape” (Jerz). The space of the original *Adventure* was based on the Bedquilt Cave, which is part of the Mammoth Cave System in Kentucky. Thus the adventure game genre has its roots in a simulation of an actual place, which incorporated puzzles and fantasy elements.<sup>32</sup>

However, the space of adventure games has also undergone extensive transformations, mainly because the technologies of representation have also evolved. From text to 2D illustrations to navigable 3D spaces, technology has reshaped the way that space is designed in adventure games, as well as many other videogame genres (Fernández Vara, Zagal, and Mateas).

The move from text to graphics also changed the way in which the space is navigated. A room is not necessarily a closed space, but rather a specific location. For example, in *Zork I: The Underground Empire* the area around the white house at the beginning of the game is an open space; each location around it has a distinctive description: “West of the House,” “North of the House,” “Behind the House,” “South of the House.” Rooms are usually interconnected in the cardinal directions and their intermediates (e.g. northwest, southwest), as well as up and down. The navigation in text adventure games thus consists of going from room to room: “go north,” “go south.”

---

<sup>32</sup> See Jerz for a detailed account of the similarities between the Bedquilt cave and Crowther’s *Adventure*

The advent of graphics first brought illustrations to (almost) every location in graphical text adventures (*Mystery House*, *The Hobbit*). As the visually represented space became interactive, rather than merely illustrative, it also brought about a change in navigation: the player would still go from room to room, but now it took time to get from one end of the room to another (e.g. *King's Quest: Quest for the Crown*). This has become true even of later navigable 3D spaces, such as *Grim Fandango*, *Myst III: Exile*, or *Sherlock Holmes: The Awakened*. As we will see in Chapter 10, navigating within the room also brought about new challenges, where some actions required the player character to go from one place to another within a specific time.

Given their text-based origins, the space of adventure games has several features that set it apart from other videogame genres. To begin with, the access to the space is not multilayered, as Nitsche stated of 3D spaces. Rather, the interaction is cycle-based, as we saw in Chapter 2. There is only one input at a time—either writing the text, or moving and clicking the mouse. The player cannot move her character and talk to someone at the same time. Each action has its own separate command and is responded with one specific output, following a conversation model.<sup>33</sup>

Adventure games are the textbook example of games of progression, as we have already discussed. The milestones in the story are also associated with specific locations of the game. However, adventure games can encourage exploration, which also becomes part of the story of the player navigating the space. We have argued before that a more complex simulation encourages exploration, because there is more for the player to do.

---

<sup>33</sup> Some text adventures can list several different actions in one command, e.g. “open the box and take the flashlight.” Each action will receive its corresponding output, e.g. “The box is open. There is no such thing.” A command such as “Take all” will have the player character take all the objects that can be picked up in one room, substituting consecutive “take + object” commands for a single one.

Spatial exploration in adventure games means not only figuring out the layout of the space, but also exploring all the objects and how they work, and talking to all the characters who inhabit the game space. This exploration contributes to the restoration of behavior—by learning how the simulation works, the player can anticipate better how to perform properly in the context of game and thus advance the story.

The following sections will be devoted to specific design devices related to space within the adventure game genre. These devices aim at promoting the restoration of behavior—in Chapter 5, we saw how restoring the behavior in adventure games meant completing the events of the story. Thus spatial design is fundamental in both encouraging this restoration and incorporating the story into the game. There are two basic categories into which these devices fall: navigation design and inscribed stories.

#### **7.4.1. Navigation as Gameplay and as Story Device**

The previous chapter introduced that navigating the space itself may be a challenge, which can take several forms. To begin with, puzzle-solving can regulate the exploration of the space if the player has to solve the puzzles to access new locations. Finding one's way, as well as mapping the space, can also become a challenge, by having the player work out the relationship between locations in the game. As we argued above, navigation can also help establish an order in which the player has to traverse the game, thus creating a structure for the events of the story to occur. This order, combined with how puzzles can regulate the access to specific areas of the space, is a basic strategy to combine game design and story in adventure games.

Let us focus on the design of spatial challenges. In his seminal book *The Image of the City*, Kevin Lynch discusses the “imageability” of the city, the quality to evoke a

strong image of a physical object (9). If we apply the concept to space, imageability refers to the capacity of the observer to apprehend the space, to make sense of it, and to establish the relationships between the different elements in the urban environment. Videogames must be able to evoke a strong image too, although the space of the game is not physical—it is not only designed, but also not bound by physical constraints. This liberation from the physical world allows videogames to create new kinds of navigational challenges (Fernández Vara, “Labyrinth and Maze”). Adventure games, in their historical evolution, have made a very good use of imageability as the foundation to their navigational challenges.

#### 7.4.1.1. Mapping

One of the basic strategies to play an adventure game, particularly in the case of text adventures, is drawing a map of the space as it is being navigated. Certeau cites Linde and Labov to discuss the dichotomy between the map and the tour as two basic ways in which to represent the space.<sup>34</sup> The tour is the description of the experience of navigating the space (“Go to the left until you get to the hairdresser’s on the corner, cross the street and go right until you reach the movie theatre”). The map is a representation of the space where each location is represented in relation to another (“The movie theatre is next to the cafe, both are opposite the ice-cream shop”).

Making a map of a space is the most elemental way to apprehend it: a map is a physical representation of the space from a different point of view, which allows the observer to figure out the relationships between different locations. The player usually

---

<sup>34</sup> Certeau is quoting the work of Linde and Labov on the descriptions of NY residents of their apartments.

experiences the space as a tour, so making a map is a way to understand the spatial relationships within the space—it helps “imageing” it, in Lynch’s terms. Going back to the types of insight, mapping is a clear case of selective combination, in which the player joins different pieces of information (each specific location) into a new one (the game map). The twist of mapping videogame spaces is that, instead of making a mental image of a physical space, the player has to create a physical image of a virtual space.

Mapping the space is essential when playing text adventures in order to traverse the space. Each room is connected with other rooms, and the player has to go from one room to another first, to figure out what is in each room and how they connect, before even starting to think about solving puzzles. The advent of navigable spaces represented with 2D and 3D graphics makes the space of adventure games more imageable, because it is easier to observe the connection between one location and the next.<sup>35</sup> The player could also observe the player character exiting one screen and going into the next, establishing an immediate and continuous representation of the space. This gave way to an interesting paradox—while navigation in graphic adventure games seemed to have become more fine-grained, and introduced real time as a design device, it also seemed to reduce the number of locations in the game. The space of point-and-click adventure games also tends to be smaller than text adventures—having to create backgrounds and animations for each space also limits how many locations a game may have. There is literally less room for exploration, whereas the interaction seems to focus on interacting with objects.

---

<sup>35</sup> Some graphical text adventures such as *The Hobbit* had illustrations for some rooms, not all of them, so that did not quite help navigation much.

Plotting a map of the space has been abandoned as a practice in current point-and-click adventure games, although some adventure games may still present certain specific areas that need the player to grab pen and paper in order to figure them out. *In King's Quest V: Absence Makes the Heart Go Yonder*, the player has to cross a desert. But King Graham, the player character, cannot walk for very long without drinking, so the player has a limited number of moves before Graham dies of thirst. This forces the player to map the space, and have the player character die multiple times until she finds the way out of the desert. Navigating the desert becomes a trial-and-error puzzle, combined with a navigation challenge.

#### 7.4.1.2. Fragmented Spaces

Some text adventure games presented a sprawling space that the player had to explore. The dungeons of *Zork*, the town of Rockvil, South Dakota, in *A Mind Forever Voyaging*, are textual versions of what the *Grand Theft Auto* series or *The Elder Scrolls: Oblivion* did graphically almost two decades later—a vast open space that the player can walk around in.

Large extended spaces can be rather daunting to figure out, however. The *Grand Theft Auto* games come with a map both in the package and in the game, in order to support navigation. *A Mind Forever Voyaging* came with a map of the town—the game is too complex to have to figure out the space without a guide. However, it is still up to the player to figure out how it becomes a series of interconnected locations (rooms) in the game.

Adventure games have thus provided ways in which it is easier for the player to apprehend the space by segmenting the space. Space segmentation is a recurrent design

device used in order to manage gameplay (Zagal, Fernández Vara, Mateas). Gameplay management are the strategies to control the performance of the player, or at least to restrict her performance so that she is able to restore behavior. One of the most basic ways to manage how the player accesses the space is dividing it up into distinct areas. The design of the game can prevent the player from accessing all areas at once, and require her to figure out an area before going to the next. This was the strategy followed in *The Secret of Monkey Island*—the player starts on Meleé Island, where there are nine different locations. At the beginning of the game, the player can only get to four of them (the village and lookout point, the clearing, the fork, and the bridge). The rest of the areas are accessible after solving specific puzzles. Thus, by opening up the space bit by bit, the player finds it easier to get familiar with the space.

A more common strategy, which spans both text and graphic adventures, is making the distinct areas separate. Each area has a specific set of goals; once they have been completed, the player moves on to the next area, usually with no possibility of coming back. There are many instances of this: in text adventures, we have *Plundered Hearts* and *Trinity*; in point-and-click, we have the *Space Quest* and the *King's Quest* series; more recently, there is also *Fahrenheit / Indigo Prophecy*.

Segmentation of the space allows controlling where the player will be, which in turn allows control over how the story of the player unfolds. It is not coincidental that each area usually coincides with a different “chapter” of the story, at times explicitly. This is further evidence of how the performance brings together story and game: the player must perform in a certain area of the game, which also corresponds to a specific segment in the story. *The Secret of Monkey Island* shows a title card announcing “Part I:

The Three Trials,” “Part II: The Journey,” “Part III: Under Monkey Island” and “Last Part: Guybrush Kicks Butt.” Each chapter takes place in three separate areas: the first and last part take place in Melee Island, the second part takes place on a ship and the third, as the title indicates, in Monkey Island itself. *Fahrenheit / Indigo Prophecy* divides its action into forty-five chapters; as the player completes the first chapter, new chapters will open up. Some chapters are presented in clusters of two or three, so the player can complete the episodes in each cluster in any order. Each chapter corresponds to a specific location; the player cannot move from location to location within the same chapter.

By segmenting the space and regulating where the player can go, game design is also establishing the flow of events and actions in the story. Narrative and spatial segmentation go together—since the movement of the player in the space generates the player’s story, when the game design controls where the player goes it also narrows at what part of the story the player can be. This also means that the restoration of behaviors is segmented, not only from puzzle to puzzle, but also from area to area of the game.

Player’s history can also be determined by the number of objects in the space and the possible interactions with them. A rich simulation offering a wide range of interactions encourages exploration, so that the story of the player can diverge greatly from the ideal playthrough of the game. This is not a negative aspect; on the contrary, it is one of the most appealing types of gameplay that adventure games can foster. On the other hand, players can also lose sight of the goals of the game, and adventure games eventually have a finite number of ways in which they can be explored. As we saw in Figure 12 in Chapter 5, while the player is exploring the simulation, she is not restoring the behavior expected in the game.



Thus adventure games have eventually opted for giving clear goals, limiting the number of objects in the space to those the player will need in order to complete the game—yet another shift towards higher levels of abstraction. *Myst* and the games in its style (e.g. *Syberia*) provide lavish visual representations of the space, but not many objects to interact with. The pleasure of exploring the possible actions and discovering the space was substituted by high-resolution graphics and figuring out how the world works. As we saw in Chapter 5 (Figure 15), text adventure games such as *Zork* and modern point-and-click games such as *Myst* are the two opposite ends of the spectrum—from a space of possibility, to a space of lush constraint.

#### 7.4.1.3. Labyrinths, Mazes, and Other Navigational Challenges

The navigation of the space can also be playful and challenging, while remaining pleasurable—playgrounds and hedge mazes are two non-digital examples. Labyrinths and mazes are prototypical navigational challenges in architecture. Both terms have been used interchangeably, although a distinction between them has been made by the likes of Umberto Eco and Hermann Kern. Labyrinths are unicursal, a single path that folds within itself. The winding path turns back and forth in order to disorient the navigator, but there is no getting lost. As long as the navigator advances, she can get to the exit (or the center of the labyrinth and then back). On the other hand, mazes can be navigated in multiple ways, and only a few of them will lead out of it. There are both branching ways and cul-de-sac, and the navigator is expected to get lost in them (Eco 80-1; Kern).

Labyrinths and mazes structure navigation, without being completely unfathomable. “Complete chaos without a hint of connection is never pleasurable” (Lynch 6). As in the case of puzzles, it has to be possible for the player to make the

connections between different locations and figure out how to navigate the space. Thus the process of imageing the space is satisfying, since making sense of a state of affairs is a challenge and a pleasure. Navigating a space that is difficult to “image” and that is meant to confuse the navigator, can be enjoyable depending on a series of conditions (Lynch 5-6):

There is no danger of being trapped: The player inhabits the virtual space, but she is not physically there, so there is no danger of physical harm. The feeling of safety in exploring the virtual space is part of its attractiveness.

The confusing navigation must be restricted to smaller areas within a coherent whole: Making navigation difficult is a feature that games can take advantage of, because it poses a challenge to the player. Given the sense of security that the player has, games can afford to increase the difficulty of navigating their whole space. It is more frequent, however, that labyrinths and mazes are restricted to specific areas of the game, such as the mazes in *Adventure* with their “twisty little passages.” The possibility of creating impossible spaces also encourages creating new navigational challenges. We will deal both with *Adventure*’s mazes and impossible spaces below.

The labyrinth can be mastered and understood over time: The final apprehension of the space is what constitutes the source of pleasure, since it corresponds to the moment of insight when solving a puzzle. Navigation can be a challenge, but it has to be a surmountable one—finding out one’s way in the labyrinth is the expected performance of the player. Once the connections between locations have been made sense of (e.g. the space has been completely mapped out), navigation stops being a challenge.

These conditions make labyrinths and mazes pleasurable, because in the end they

are akin to puzzles. Rather than having a moment of insight, the player has to pay attention to how the space is laid out, and also come up with strategies to navigate it. Designers can also come up with ways to complicate the navigation, and encourage the player to devise ways to overcome the challenge.

A famous example of how to complicate the navigation in a maze is the original *Adventure*. One of the mazes consists of a set of rooms that are described in similar terms, but changing the phrasing slightly: one part is described as “a maze of twisty little passages, all different,” and the next is “a maze of little twisty passages, all different.” As Montfort points out (*Twisty Little Passages* 90), the game rewards the player who reads the description carefully. The pirate’s maze in the same game, on the other hand, has exactly the same description for every single room (“You are in a maze of twisty little passages, all alike”). Roberta Williams, designer and writer of several graphical text adventures and the *King’s Quest* series, describes her strategy to map the pirate’s puzzle in *Adventure*:

The confusing thing about most mazes is that all places look or seem alike. So, no matter what direction you go, you don’t know where you are, where you’ve been, or where you are going. That makes mazes impossible to map, unless you can make each place look different. The answer to this problem is really quite simple: drop objects along the way. When you first enter a maze, drop an object there, right away. Then you will know that is the place to exit from the maze. Get as many objects as you can and drop one each place you go. Then try going all the directions from each place and soon you will start seeing objects that you have dropped. Draw a map marking down what you dropped, the direction you went, what you dropped at the next place, and what direction you went from there. After a while, when you start running into previously dropped items, you should be able to look on your map and see how you got there.

Montfort and Williams propose two different ways to traverse these mazes successfully. Although the mazes in *Adventure* may also be unusual even for text

adventure standards, they are also taking advantage of the properties of the medium to design the challenge, namely the textual descriptions of each location.

#### *7.4.1.3.1 Impossible Spaces*

Videogames also make it relatively easy to design spaces that are not physically possible (Fernández Vara, “Labyrinth and Maze”), which makes them more difficult to image. Virtual spaces can escape the constraints of topology, and create spaces that we could not experience in the real world (Novak). Thus, the space in a game can be designed so that it does not map to a real space.

A space that would not have a physical equivalent can be difficult to image, in Lynch’s sense. It also promotes the restoration of a set of behaviors that the player has no real referent for. Thus the game invites the player to come up with new behaviors that she could not possibly restore outside of the game.

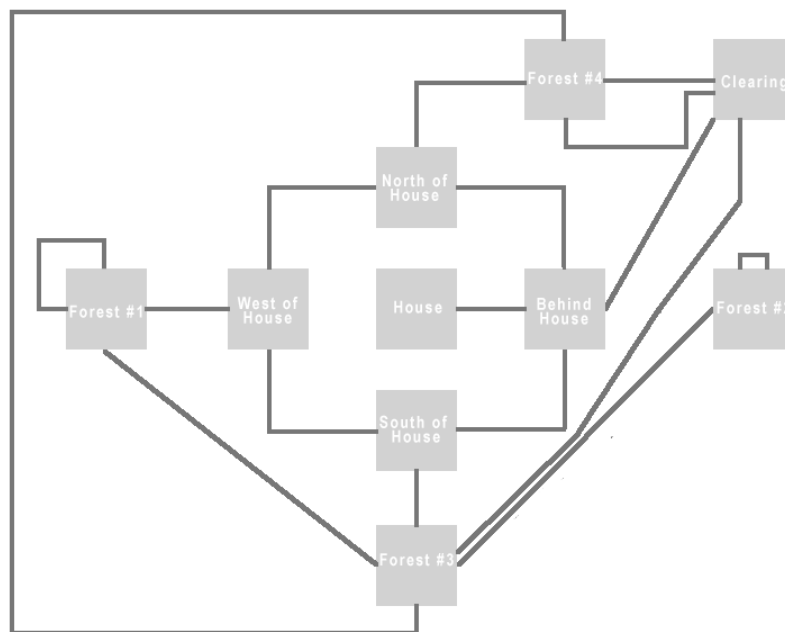
Spaces represented in text can be flexible, unstable, and not possible in the real world (Fernández Vara, “Labyrinth and Maze”). In text adventure games, the programmatic properties of each room allow designing physically impossible spaces. Each room is a node (“the smallest meaningful space into which a player’s character can fit”) rather than a visual space:

The arrows on the map need not be bi-directional – north from room A may lead to room B, but south from room B could lead to room C. Arrows need not connect different rooms. North from A may lead back to room A. Because rooms are just another kind of object, they can be picked up, carried around and placed inside one another – or, indeed, inside themselves. Arrows can be changed dynamically to point at different rooms.” (Bartle, “Making Places” 159)

Bartle is proposing two things here: on the one hand, there can be relationships

between rooms that do not correspond to a possible configuration in the real world. On the other, he is describing unstable spaces, which change as the player navigates them. Adventure games present archetypical examples of both.

The map of *Zork*, as well as the maps of many text adventures, is not supposed to be a topographical representation, but a map of relationships between rooms. Below we see a map of the initial area of the game.



*Figure 16: Map of the initial area of Zork (excerpt).*

We see how going north of forest #1 takes the player back to the same place, and how going north of forest #4, which is north of the house, takes the player to forest #3, which is south of the house. Given the fragmentation of the space into rooms, the intricate connections make the space not consistent in physical terms.

The instability of space is not unique to text adventure games, however. 2D

graphic adventure games such as *The Secret of Monkey Island* fragmented the space into screens, which can also be considered nodes. This fragmentation also allowed creating connections similar to the ones described by Bartle. For example, in the last section of *The Secret of Monkey Island*, the player has to navigate the underworld of the island in the title with the aid of a magical item, a zombie navigator's head. The underworld is a procedurally generated space, where the connections between rooms change randomly whenever the player chooses the wrong path. It simulates a maze precisely by creating an unstable space, where going the right may make you appear at the right of the next screen, and then if you go back left, you appear in a different location altogether. The game is designed so that the player cannot get through without the navigator's head, which turns to the correct exit in each screen. The fragmentation and inconsistency of the space also contribute to creating the experience of being lost, because it is not possible to "image" the space.

The continuity of 3D spaces, however, phased out with the inconsistency of the space. The three-dimensional space is represented continuously, trying to assemble a topologically consistent space, with little room for navigational challenges.<sup>36</sup> On the one hand, considering that the player needs pen-and-paper in order to complete a game is a somewhat old-fashioned notion. On the other, the absence of labyrinths and impossible spaces also reduces the types of puzzles that players can solve in adventure games.

#### **7.4.2. Inscribed Stories**

---

<sup>36</sup> The recent *Tales of Monkey Island: The Launch of the Screaming Narwhal*, presents a couple of treasure-finding puzzles also based on the fragmentation of the space: only by choosing the right path will the player arrive to the treasure. We can attribute this to the 'revival' spirit of the game, where it is trying to echo the puzzles of the original game series nine years after its last installment.

The space in adventure games is eminently at the service of the story. As we have seen above, advancing in the space is almost inextricable from advancing in the story, and very often gameplay is based on discovering the story of the space itself. The stories inscribed in the space, by providing more information about the fictional world, also serve as the type of information needed to solve puzzle and thus restore behavior.

Adventure games present very specific devices to inscribe the story in the space, which have been handed down from their textual origins. Although they are not unique of this genre, inscribed stories are key to understanding how the space is not only the setting of the story, but the story becomes part of the space as well. There are two basic types of inscribed stories in adventure games: object descriptions and readable documents.

#### 7.4.2.1. Descriptions

In text adventures, every input was responded with a textual output, which indicated whether the action had been successful or not. This textual output usually described the object or person (after “examine X” or “look at X”), or the reaction of the object to the action the player had just tried. These descriptions are information that is attached to the object, which extends the amount of information that can be elicited from the object. Whenever a new location was entered, a short description of the space would be given in a line or short paragraph.

That is why it is very common to start text adventure games with the description of the location. There may be an initial prologue, as in the case of *Trinity*:

Sharp words between the superpowers. Tanks in East Berlin. And now, reports the BBC, rumors of a satellite blackout. It's enough to spoil your continental breakfast.

But the world will have to wait. This is the last day of your \$599 London

Getaway Package, and you're determined to soak up as much of that authentic English ambience as you can. So you've left the tour bus behind, ditched the camera and escaped to Hyde Park for a contemplative stroll through the Kensington Gardens.

### **Palace Gate**

A tide of perambulators surges north along the crowded Broad Walk. Shaded glades stretch away to the northeast, and a hint of color marks the western edge of the Flower Walk.

The first two paragraphs introduce the setting where the player starts (London, at a time of turmoil, although no dates are given), and who the player is: implicitly, an American tourist who is on his (or her) last day of holidays. The location is established as the Kensington Gardens. The second paragraph describes the starting location of the player: the palace gate in the gardens. This is also the text that will be given as a response to “look,” which is examining the location the player character is in.

The description of the space first establishes where the exits to this location are (northeast and west), which is a basic piece of information that the player needs. The “tide of perambulators” refers to a large group of nannies pushing them around—they are the first obstacle the player must overcome in order to reach the exit to this area, since they keep getting in the way.

With the advent of graphics, the detailed descriptions of each location became shorter, because the player could see the location, even if it was rough. The first graphical text adventure was presented by Ken and Roberta Williams as *Hi-Res Adventure #1: Mystery House*. The “high resolution” are line graphics, which allow the player to recognize the different parts of the house. The text descriptions of each room still give the vital information, starting with where the exits of each room are. For instance, one of the rooms is a “dusty library,” because one can hardly tell it is a library (there are few books



on the shelves, a fireplace, a sofa and a table), and 1980s “high-res” was still far from rendering dust. Thus, the text (even if it were only two words) allows the player learn more information about what she can see.

Better graphics led to dropping location descriptions altogether, although the object descriptions were kept as a way to extend and call attention to specific details about the object itself. This was due to the low-resolution graphics, which were still too blocky to provide the kinds of subtle information that the player may need. If we look at the graphics of early Sierra adventure games, such as *Space Quest: The Sarien Encounter*, or the original *King’s Quest*, they are still very blocky, so that at times a textual description is needed in order to identify the object that is being examined. With the reduction in the number of verbs, the descriptions were reduced. However, clicking on an active object will still prompt a textual response, except in *Myst*-like games (more on those below).

The descriptions of each object and space can transform an object from a prop into an evocative narrative element, because they tell a fragment of the story, usually the pre-history. The episodic games of *Sam & Max* use this to great effect. In *Sam & Max Season One: Culture Shock*, the verbs are implicit, so that clicking on a character will start a dialogue; if it is an object, it will either be described or be picked up. Clicking on an object can also start a conversation—a mailbox on the street triggers the following dialogue:

Sam: “Tampering with the mail is a federal offense, punishable by fine and imprisonment.”

Max: “As we found out after the incident with the garden hose. Ah, memories.”

This piece of dialogue transforms a prop in the space into a perfect evocative narrative element—they are talking about something that happened in the past, but they are not giving away too much. It is still up to the player to imagine what happened in that incident with the mailbox and the garden hose.

The most important role of descriptions, however, is being the source of information that the player needs in order to solve a puzzle. For example, in *The Secret of Monkey Island*, the description of one of the pots in the kitchen is “Someone cooked a head cheese in this.” The apparently nonchalant description is actually a hint—the player has to use that pot as a helmet in a later puzzle, so it is convenient that it seems to fit a “head” already.

Although the implication so far is that the inscribed story is usually textual, there may be instances where that inscription can be visual, or even aural. *Loom* exemplifies this very well: whenever the cursor hovers over an object, a close-up of it will appear on the corner. Clicking will prompt the description itself. In the case of certain objects, they also have a four-note melody associated with them, which plays out when the object is clicked. That melody usually refers to one of its properties, and when the player plays it while pointing to a certain object, that property will be applied to the object. For instance, a bottle of green dye plays a melody that turns things green; a spinning wheel plays a melody that turns straw into gold.<sup>37</sup> Thus the melodies are actually actions, which the player can learn and use in the world.

Object descriptions are not the only way to inscribe the story in the space—in-game documents and records also serve a similar purpose.

---

<sup>37</sup> This seems to be a reference to the Brother’s Grimm fairytale “Rumpelstiltskin.”

#### 7.4.2.2. Documents and Records

As we have already established, the adventure games that implement their fictional world using a higher level of abstraction also tend to have less verbs, resorting to context mechanics more often. These games usually do not let the player choose a specific action, but rather provide it contextually. That was the case of *Myst* and the games that imitate it, where the higher level of abstraction meant doing away completely with the inscribed descriptions of the objects. However, *Myst* manages to include the story and clues in the space in the form of in-game documents and texts.

The information about the world can be accessed directly by reading about it, rather than interpreting the indexes in the world. Documents and records create a hypodiegetic level, which tells the pre-history of the fictional world, but it is a level that the player cannot interact with or affect. They provide information that extends the space and helps player's performance at the diegetic level.

*Loom* is a clear example by using close-ups in order to display the inscribed information. By clicking on a tombstone, for example, the player can read what is written on it, rather than having the player character "read it" to the player. However, the player character will still make a remark about the most relevant phrase for the game after exiting the close up.

More famously, *Myst* told the story of the fictional world through documents, through a collection of books in the library. Each book is a diary, which tells how each island ("ages," as they are called in the game) was constructed, and also provide clues about to solve the puzzles that grant access to the other islands. There are also videos embedded in the books—the two brothers, Achenar and Sirrus, have been imprisoned in

two magical books, and ask the player to help them. In the Channelwood Age, the player can find a video player of sorts, which shows videos of the two brothers bickering, and gives some insight about why they may have been imprisoned by their father.

*Gabriel Knight: Sins of the Fathers* provides another example of how documents (more specifically books) serve as a way to both inscribe the story and provide information relevant to the player. Documents are vital in this detective story, since the main mechanic of the game is gathering information to solve the Voodoo Murders that Gabriel Knight is investigating. Conveniently enough, Gabriel owns a bookshop, which provides him with reference material too. By checking a dictionary, the player gets the translation to a few terms in German. These terms help understand a poem (actually a riddle), which explains how to open up a clock that hides a letter in a secret compartment.

*Gabriel Knight* also provides a novel example of inscribed stories. Early in the game, the player can find a tape recorder, which Gabriel will use to record all his conversations with other people. These records are available for the player to check at any time, since those conversations will provide vital information to solve the case. The recordings, rather than inscribing the pre-history of the game, are inscribing the player's history, and making it easy to re-examine.

In a way, documents in adventure games are the story equivalent of the mini-games in game design. More often than not, they are akin to micronarratives, telling a shorter set of events during the game. These events are usually relevant to the story of the game—it can be the pre-history of the game, as in the case of *Myst*.

## **7.5.Conclusion**

Performing in the simulated space brings together game and storytelling, which makes it

the most relevant game element to tackle when studying stories in games. In the particular case of adventure games, space is the foundation of the game: the first requisite to encourage exploration, the origin of a variety of challenges, and the setting of the story. The objects in the space are not only props in it, but also partake of the story of the fictional world, since fragments of the story can be inscribed in them. This chapter has been but an overview of the possibilities of studying space as the connecting element between game design and story. Adventure games provide us with some of the clearest examples of this relationship; applying similar terms to other genres may be a way to find similar design devices, or perhaps finding a design gap.

## CHAPTER 8 PLAYER CHARACTER

### 8.1. The Player in the Simulation

This chapter and the following explore characters in adventure games not only because they are a defining feature of adventure games, but also because characters are also a defining element of the story—they are part of the *existents* of the story (Chatman). First, we will define what “character” is, and how the concept applies to videogames. Then we will focus on adventure games and the different features of the player character: its functionality, how it shapes the relationship with the player, its identity in relation to the fictional world, and how the design of the player character provides cues to restore behavior the simulation. One of the key questions in this section is to what degree a player character can have a personality that brings together the actions of the player with who the player is supposed to be. To explain how the design of the game can reconcile both, we will resort to Janet Murray’s concept of “scripting the interactor” (Murray, *Hamlet on the Holodeck* 79), and apply it to the process of restoration of behavior.

### 8.2. Defining Character

In narrative terms, a character is “an existent endowed with anthropomorphic traits and engaged in anthropomorphic actions; an actor with anthropomorphic attributes” (Prince 12, based on Chatman). The anthropomorphic qualities of a character in a fictional world seem to be a device to involve the audience in it. In theatre, for example,

the audience feels for the characters, is involved in the action of the play because there are other human beings in it.<sup>38</sup> Even in the case of stories told with abstract shapes, we tend to personify those shapes and endow them with human-like behavior—the protagonists of Abbott’s novel *Flatland* are squares, triangles whose anthropomorphic qualities derive from human-like behavior. Stories do not happen to things, but to people.

Characters are also defined as “an existent engaged in action” (Prince 12). That is, characters are also agents in the story, they perform the actions that create the changes of state. This is the quality that allows characters to have a double role as game entities—their attributes and actions are also part of the game. The following sections will deal with the different aspects that make up the duality of characters as existents of the story and game entities. We will tackle the player character separately from non-player characters (NPCs), because they are different elements in the game design—while the player character is the entity the player controls in the simulation, non-player characters are simulated and controlled by the computer.

One of the gaps this chapter is trying to address is how characters are integral to game design. Steve Meretzky makes a distinction between *characterization* (what is observable about the character) and *character* (what is underneath, the essential nature of someone) (Meretzky, “Building Character”). For the sake of having clear terminology, we will refer to this second concept as the *personality* of a character, which also refers to how a person may behave according to who (s)he is and how (s)he thinks (Isbister 35-6). The personality of the characters is usually dealt with rather superficially in games,

---

<sup>38</sup> See Aristotle for a classical approach, Garner for a more recent phenomenological account for this empathy.

precisely because the concept is quite complex.

Meretzky notes that works dealing with character tend to focus mostly on characterization, i.e. what the character looks like, and then resort to Hollywood screenwriting formulas to construct character, such as establishing character goals and defining traits, such as weaknesses and strengths (see Adams and Rollings; Krawczyk and Novak). Imitating screenwriting manuals also means that these previous works on videogame characters tend to be very prescriptive, as if there was a unique formula to design characters. In general, these approaches are based on other media, such as comics or film, which seems incomplete—they are only addressing the aspects of character that are represented, but not simulated. Designing a character for a game is not the same as writing a character for a novel or film, even though both are related.

A character has rules attached to it. The player character is associated with the game mechanics, which provide the player with the means to perform in the simulation. The NPCs are associated with the rules of the simulation, which establishes how they have to behave and respond to player's input. These are the game design aspects that are usually overlooked.

Petri Lankoski and Staffan Bjork have acknowledged the problems of relying mainly on traditional narrative formulas and visual representation, and have proposed a character-driven game design method, where the personality becomes an important part of the gameplay. Inspired by dramatic writing, the method focuses on how writing can inform the design, rather than shoehorning it into games. The drawback of their proposal is the opposite of the Hollywood formula technique—it is based on a specific case study (and adaptation of *Les Liasons Dangereuses*), so its results are difficult to generalize.



Katherine Isbister's work also focuses on characterization more than designing personality, her approach to studying videogame characters is based on social psychology. Her study deals with the relationship of the player with the characters of the game, based on how social relationships are established. This approach to designing characters, however, overlooks how characters are also a story element, and how social interactions in the game can be part of the simulation.

The approach of this chapter and the following is not to propose a design / writing method for characters, but rather to survey the different design elements that help construct the personality of a game character in an adventure game. The focus will be on the design of the characters as part of the simulation, and how they integrate in the fictional world, rather than on basic dramatic writing guidelines that have been covered extensively elsewhere.

The novelty of the current approach is taking the standpoint of performance, talking about character design in terms of the interaction with the player and how it behaves in the simulation, rather than how the player perceives the character. In the case of the player character, there is an extra layer of complexity, because it also constitutes a specific point of view with respect to the fictional world.

### **8.3.What is a Character in a Videogame?**

Before dealing with the specific case of adventure games, we should first define what we understand as characters in videogames. This is a wide topic that has been dealt with by some of the authors quoted above (Isbister, Lankoski), as well as Rune Klevjer, who focuses on the relationship between player and avatar ("What is the Avatar?"). These works define player character/avatar, and non-player character, but a basic

definition of what a character in a game is seems to be taken for granted. This is why we need to provide our own definition of what a character is in a game, before making the distinction between player character or NPC.

A character in a videogame is an entity in the simulation, which is an agent in the actions and / or a source of information. It is a token with which certain rules of the game are associated, e.g. movement, speed, or health points. A car in a racing game might be considered a character. However, it does not seem appropriate because characters also have to be anthropomorphic, just like characters in narrative. In order to qualify as a character, the entity must have human-like features or behaviors, and be animated (i.e. they seem to have a will of their own). A videogame character is therefore an anthropomorphic agent in a game, who performs actions associated with human behavior.

The anthropomorphic qualities are an important index of the degree to which a videogame is story-driven. A car in a racing game, for example, could be thought of as a character, but it is certainly more difficult to talk about it in terms of story (unless the car acquires anthropomorphic qualities, as in one of the videogames based on Pixar's *Cars*).

An anthropomorphic game entity can also be an agent that participates in the events of the story. Unfortunately, the relationship between their actions in the game and in the story is often rather tenuous. For example, the character may appear in a cut-scene scene where its actions have nothing to do with they can do with the game, i.e. their represented behavior is different from their behavior in the simulation. To take an example from fighting games, in *Soul Calibur 2* the characters will taunt each other verbally before a fight, and then the winner will brag when it is over. During gameplay, however, talking or taunting are not available actions. This is also an example of how the

segments of the story in the form of cut-scenes are, in a way, rewards to the player for achieving a goal (Juul, “Games Telling Stories?”).

In story-driven games, the overlap between the actions of the character in the story and in the game has to be practically complete. In RPGs the characters talk, fight, travel, exchange goods, and explore, and both are represented and simulated actions. This overlap is also present in adventure games, since the list of actions is more varied than in other genres. The contrast is even more marked if we compare them with fighting games, for example, where all the actions are physical. In the case of adventure games, some cut-scenes may be used occasionally to illustrate verbs that are not part of the list. Mini-games, as we saw in Chapter 6, are also a device to bring more specific or nuanced activities, since they change the level of abstraction of the simulation.

Characters, like the space, may also have a pre-history, which has shaped them the way they are. In the case of the player character, this can create a gap between character and player, since the player does not know “who she is” in the story. Some of the devices used to erase that gap have been making the character a “blank slate,” with no past, or have the player find out who the player character has been. Using amnesia as a device is too trite nowadays, although one of the first games to use such premises was a text adventure, *Amnesia*, by Thomas M. Disch.

Creating a videogame character is not only giving it certain looks and way of speaking, which is what most videogames do. It is also designing the rules attached to it, the behavior of that character, making that entity / existent an individual that is recognizable as human (or human-like). Following prescriptive formulas as proposed by Adams and Rollings probably results in shallow characters, repetitive and predictable.

After this brief overview of what the term “character” means in the context of videogames, we will continue the discussion with relation to adventure games. We will deal with the player character first, because it poses different design problems from NPCs, and establishes the relationship of the player with the simulated world. NPCs will be covered in the next chapter.

#### 8.4. The Player Character in Adventure Games

The player character is one of the defining features of the adventure game genre. It is the bridge between the player and the simulation; it is the game entity the player uses to interact with the simulated world. Adventure games have at least one character: the player character, who performs the commands of the player in the simulation. As we will see below, some adventure games have more than one player character, although this is not very common.

The very first adventure game, *Adventure*, had a peculiar way of introducing its player character, as Buckles and Montfort have noted. If the player asks for instructions at the beginning of the game, she will read the following:

Somewhere nearby is Colossal Cave, where others have found fortunes in treasure and gold, though it is rumored that some who enter are never seen again. Magic is said to work in the cave. **I will be your eyes and hands. Direct me with commands of 1 or 2 words.** I should warn you that I look at only the first five letters of each word, so you'll have to enter “northeast” as “ne” to distinguish it from “north”. You can type “help” for some general hints. For information on how to end your adventure, scoring, etc., type “info”.<sup>39</sup> [my emphasis]

---

<sup>39</sup> This is from version 2.5 by Don Woods. For a more detailed account between Woods’ and Crowther’s versions, see Jerz.

“You” seems to refer to the player, the person who is interacting with the game. But who is “I” in the first quote? Buckles argues that it is the narrator, as if it were the Dungeon Master of a table-top RPG (Buckles, “The Computer Storygame *Adventure*” 141-2). It could also be interpreted as the game itself—the help text also uses “I,” but the pronoun is missing outside of these extradiegetic texts. The paragraph refers to how the game works, how the player should input the commands on the game. However, immediately after that, the first output of the game tells the player: **“You are standing at the end of a road before a small brick building. Around you is a forest. A small stream flows out of the building and down a gully.”** [my emphasis]

In this case, “you” seems to refer to the player again, although now it is not the interactor any more—it is whoever is in the simulation, presenting it to the player. The “eyes and hands” entity does not refer to itself in first person, but in the second, in order to involve the player. This apparent split between the player and the entity that describes the simulation is the perfect cue to ask what the player character is, what it does in the game, and finally, who that character is. In the following sections, we will address each of these questions.

#### **8.4.1. Functionality of the Player Character**

The player character is a game entity that is controlled by the player, and performs the commands of the player in the simulation. It is the “hands” that *Adventure* referred to. This entity also obtains information about the simulation, and transmits it to the player—it is the “eyes” that examine the world. The descriptions of the objects, which were dealt with in the previous chapter, are usually given from the point of view of the player character. The character the player controls is also the interface between the player

and the other characters in the simulation (Isbister 203).

Adventure games have a large and varied catalog of player characters: from detectives (*Deadline*, *Sam & Max Hit the Road*, *The Lost Files of Sherlock Holmes: The Case of the Serrated Scalpel*, *Hotel Dusk: Room 215*, *Indigo Prophecy / Fahrenheit*), space janitors (*Planetfall*, *Space Quest* series), ladies in distress (*Plundered Hearts*), pirates (*Monkey Island* series), novelists (*Gabriel Knight* series), adventurers (*Indiana Jones and the Fate of Atlantis*, *The Last Express*), and even humans turned into cockroaches (*Bad Mojo*).

The player character can also be considered the protagonist of the story, since it provides the point of view within the simulation. In the case of multiple player characters, the player will be offered different points of view, or at least different ways to access the world – we will elaborate on this below. In functional terms, the player character can also work as the Proppian hero, particularly in games that follow a quest-like structure (*The Hobbit*, *King's Quest* series, *Indiana Jones and The Fate of Atlantis*). The hero is assigned a mission, which becomes the overall goal of the game. The player character is the protagonist of the story, not because things happen to the character, but because the character is the main agent of the story.

#### 8.4.1.1. Player Character vs. Avatar

Throughout this dissertation, we will use the term “player character” to refer to the player-controlled entity in adventure games. “Player character” and “avatar” are normally used interchangeably, as Klevjer notes (“What is the Avatar?” 12). However, for our purposes we need to make a subtle distinction between both; each term defines different ways of participating in simulated environments and fictional worlds.

Klevjer prefers to use the term “avatar”, because he considers the avatar a “prosthetic vicarious embodiment” (12), and understands it as an extension of the player in the simulation. This concept seems useful for the genre that he is concerned with, action-adventure games (mainly First-Person Shooters).

In the case of adventure games, the relationship between player character and player in adventure games is more detached, and more difficult to account for in terms of embodiment. We will prefer the term “player character” in the context of adventure games, as a way to highlight that separation between player and character.

#### 8.4.1.2. The Problem of Embodiment

One of the main arguments against considering the player character as embodying the actions of the player is outlined by Montfort, who proposes that the player character is more of a “vehicle” that the player steers (“Fretting the Player Character” 140-1). He argues that the player does not “play” the character, as players may do in table-top RPGs (where adventure games also have their roots), or in a theatrical performance. Playing the PC is different from dramatic play, because in Interactive Fiction (and in adventure games, by extension) the player is not really embodying the character, meaning using her physical body to enact those actions.

Montfort also asserts that the player does not have motivations as an actor would (“Fretting the Player Character” 139). This argument, however, is somewhat weaker if we look at it in terms of game design. The player is usually given a goal in the game (e.g. solve a mystery, escape from a specific place, find treasure), which can work as a motivation for both the player and for the player character. The goal and motivation of the performance is restoring the behavior, which is encouraged by the game design.

What the player character in adventure games does not usually allow is a range of play styles, or different methods to tackle the problem. The game expects the player to solve the puzzles following a set of pre-determined steps, constraining the player to act in a specific way. Going back to the performance framework introduced in Chapter 4, theatre actors can interpret the same text in different ways, depending on the goal of their performance, which also changes how the audience understands the text. In adventure games, the game mechanics (which are the equivalent of the theatrical text), dictate the dynamics very tightly since there is only one solution, therefore not giving the player much material to come up with her own performance. In games with a higher level of abstraction, where the game determines the verb to be used depending on the context (e.g. *Myst*, *Sam & Max Season One*, *Ceville*), this constraint is even more obvious, since there is not a range of possible actions to even try with an object.

Apart from the range of actions, the detachment between player and character also makes it difficult to relate the control of the player character to theatrical performance. Whereas in other genres the avatar may actualize and embody the actions in the simulation (e.g. First-Person Shooters, fighting games), the player character of adventure games follows the instructions of the player. We stated in the previous chapter, the access to the space is not multi-layered: there is only one channel to interact with the simulation, and that is through the player character.

As I have argued elsewhere (Fernández Vara, “Shaping Player Experience”), the adventure game interface has evolved from an indirect manipulation scheme, and incorporated more and more elements that bring direct manipulation to the game



interface.<sup>40</sup> The concept of direct manipulation was first described by Shneiderman. Shneiderman established the difference between using command-line input and using immediate, continuous visual feedback to understand the actions in the program. Indirect manipulation refers to text-based input, where the interactor writes a command, and then receives a reply stating whether a command has been successful. This type of interface, as Shneiderman remarks, is very prone to errors because the interactor has to wait for the computer to approve of the command or not, and there are many ways in which the command can fail (e.g. typos, bad syntax). Text adventures/Interactive Fiction all follow an indirect manipulation scheme, structured in cycles (Montfort, *Twisty Little Passages*)—all the input and output is written; all the locations and objects are described, and the player must write the commands and wait for the game to state whether the action is successful or not. Direct manipulation, on the other hand, refers to the interaction that uses a graphical interface, manipulating objects through physical actions or pointing. The point-and-click interface like the one found in graphic adventure games exemplifies how direct manipulation was incorporated in the genre.

The search for direct manipulation may be a strategy not only to improve the usability of the adventure game interface, but also to bridge the gap between the player and the character. It can be argued that the commands of the player are not really embodied, since they are typed on a keyboard. Indirect manipulation provides further evidence of the detachment between player and character, another way in which adventure games can undermine player's immersion in the simulation. Klevjer implicitly

---

<sup>40</sup> The rest of the section is a revision of my arguments in the article "Shaping Player Experience in Adventure Games: History of the Adventure Game Interface."

omits text adventures from his study, because his study is concerned with “tangibility,” meaning an interaction that simulates physical interaction, which also implies visual representation; he mentions *Adventure* precisely as one of the counterexamples to the concept (“What is the Avatar?” 119-8).

Point-and-click interfaces are an example of direct manipulation; however, it is still questionable whether using the mouse is a way of incorporating embodiment in adventure games. The mouse movement and clicking are directly manipulated by the player, but it is still the player character who was to carry out the action. For example, in *The Secret of Monkey Island*, the player first selects the action by clicking on the verb list, and then the object she wants to operate on (e.g. “open chest”), and then Guybrush, the player character, will do the action if it is an acceptable command. If the player, for example, tries to move the chest, Guybrush will reply “I can’t move it.” The player is in effect reconstructing the command line, by clicking on the objects rather than typing the commands, and waiting for an output of whether the action is successful or not. Thus, although the interface of the game uses direct manipulation, the player is still affecting the simulated world indirectly.

The player character is usually indirectly manipulated, so that it is closer to a puppet, who will walk to wherever the player has clicked. The graphical interface changed the way in which the player perceived the world and navigated it, facilitating finding what the possible actions could be (as opposed to “word fishing” in text adventures), but the way the commands work was still the same. Few games allow controlling the movement of the player character directly using joystick or directional arrows. Two of the examples of direct manipulation of the character are first forays in a

new type of interface for adventure games: *King's Quest: Quest for the Crown* (1984) was the first game with interactive graphics, while *Indigo Prophecy / Fahrenheit* proposed a new type of interface for the genre.

There are some exceptions, however, that demonstrate how some adventure games have tried to incorporate embodied action into their gameplay. There are adventure games that allow the player to affect the world directly, as in the case of *Myst*, where the player clicks on the object that she wants to interact with. *Myst* gets rid of the explicit command line by choosing the action by default for the player (e.g. pull a lever, move a piece, open a book and display it). The player character is implicit, by giving the player a first person point of view, thus conveying a stronger sense of physical immersion. Nevertheless, since the player cannot specify what action to carry out, it is also difficult to think of the first-person point of view as encouraging embodiment either. *Myst* avoids having messages informing the player that the player character cannot carry out an action by not allowing other actions. The absence of error messages is another factor in direct manipulation (Shneiderman 494), but also curtails exploring the variety of actions that other adventure games had, as was the case of text adventures. For example, what if the player wanted to take the books to read them any time, rather than having to go to the location where they are? The player is not given any reason why the books cannot be taken into an inventory.

More recently, adventure games have tried to incorporate embodiment into gameplay through the use of new interfaces. The incorporation of motion sensing technologies (Nintendo Wii, PS3) and touchscreens (Nintendo DS), encourages gestural interfaces, where the player's bodily movement is translated into an action in the

simulation. These devices expand the space of the performance to the player's space, outside of the simulation. The right physical movement in this extradiegetic sphere can have an effect on the simulation of the game.

Adventure games in these new platforms include puzzles where the player has to perform a physical gesture. For example *Hotel Dusk: Room 215* has a puzzle where the player has to actually close the DS, as we saw in Chapter 6. The Wii version of *And Then There Were None* has the player moving the Wii remote as if it were a shovel in order to remove flour from a bag. Consoles are not unique in having gestural interfaces—the online game *Bow Street Runners* has some puzzles that use mouse movement to imitate gestures, such as a swift curved movement to pick someone's pocket. In all these instances, the puzzle is getting the movement right, as we saw in Chapter 6, but how it is achieved, or whether there may be different ways to perform the movement, does not count in the interaction. This seems to bring about another paradox—by incorporating the player's bodily movement in the game, the player becomes more aware of how the movement can take place, undermining whatever sense of immersion the game is trying to promote. The shovel puzzle, for example, turns out to be quite difficult to get right, since the controller cannot detect the necessary nuance. The movement is not the same as in real life—its restoration is rendered unnatural by its mediation.

*Indigo Prophecy* (US)/ *Fahrenheit* (Europe) is an adventure game that aimed at reinventing the interface of the genre, by emphasizing many components that involve direct manipulation and using context mechanics. The player can move the player character with the left analog stick (rather than clicking to where the character should go), displaying contextual actions depending on where the player is. For example, if the

character is sitting at a desk with a computer, the game will offer several actions that can be performed in that context: pulling either drawer to each side of the desk, or using the computer. These actions are selected by moving the right analog stick, and the movements often imitate the gesture that the player character would perform. In this case, pulling the analog stick down means getting closer to the computer, and moving it to the right will open the drawer on the right of the screen. At times the gesture is somewhat simplified—in order to climb, the player must move the analog stick in the shape of a G, imitating the way in which the arms would move.

*Indigo Prophecy / Fahrenheit* also includes many puzzles based on indirect manipulation as well, where the relationship between what the player has to do with the controller and the action in the simulation seems arbitrary. In these puzzles, the timing seems to be more important than the gesture, so the significance of the physical action is lost. These design devices are *difficulty metaphors* (Juul, *Half-Real* 172-4), where a difficult task is substituted for a task that metaphorically has a similar difficulty. Fighting another character, or escaping from a floor that is collapsing, are all made into *Simon*-like actions,<sup>41</sup> where the player has to move both analog sticks in a specific sequence and on cue. This is closer to what is also known as “quick time events”, where the game tests the reflexes of the players, rather than performing the right action. By moving both analog sticks in the game controller arbitrarily (in the fashion up-up-left-down-left-right) in order to be able to perform a successful action, *Indigo Prophecy/Fahrenheit* brings back indirect manipulation, disconnecting player’s actions from the events in the simulated

---

<sup>41</sup> By Simon, we refer to the electronic game designed by Raph Baer and Howard Morrison (US patent 4207087) and manufactured by Milton Bradley. *Indigo Prophecy / Fahrenheit* uses a similar circular layout, and the same colour scheme, even if the colours are in different places.

world through difficulty metaphors.

To sum up, in terms of functionality, the player affects the simulation through the player character acting as a puppet of sorts. Traditionally, the player has commanded the player character what to do rather than acting directly, precluding embodiment as a factor in the interaction. Gestural interfaces, more often than not, allow the player to perform the action physically, rather than controlling the player character itself; thus the puzzle becomes getting the movement right, rather than finding the correct action. Thinking of the player character as a puppet is also problematic, as Montfort notes. Although the metaphor accounts for the anthropomorphic qualities of the character and how it carries out the commands and movement, “it fails to capture how the world is presented to the interactor from the perspective of this character” (“Fretting the Player Character” 140). Who the player character is and how it perceives the world are also key concepts in the design of the character. The following section will deal with the identity of the player character, and how it is established with relation to the player.

#### **8.4.2. Identity**

The evolution of the adventure game interface has affected not only how the player character is controlled, but also who that character is. While the functionality of the player character has more or less remained the same, as we just saw, the changes in the interface have affected the identity of the player character and its relationship with the player.

How the player controls the character, as well as what actions the character can perform, set the foundations of the identity of the player character. Identity in this context refers to who the player is in the fictional world. There are different elements define who

that character is; traditional approaches, such as Adams and Rollings, create the identity of the character through the visuals and the writing. However, a character is also defined by what it does, by her actions—this is true of traditional narratives, where each character type is defined by what they do (see e.g. Propp’s *dramatis personae* of the folktale, or Aristotle’s concept of unity of action and character). The problem is that most of the character actions are commanded by the player, which poses a problem when it comes to defining the identity of the player character. There is a clash between what the player may want the character to do, and what the player character is supposed to do in story. The player can experiment with the world, but in the end is not given choices about how to address the problems posed by the game. This constraint is one of the features that distinguishes adventure games from other genres; being a game of progression means that there are certain set actions that the player is expected to do, or otherwise the restoration of the proper behavior will not happen.

#### 8.4.2.1. Player Performance: “Scripting the Interactor” and Restoration of Behavior

The dynamics of adventure games are very constrained, since in order to traverse the game, the player must restore very specific behaviors. Although it is also true that adventure games often encourage exploration, and that there may be puzzles that are not necessary to solve in order to complete the game, in the end, the player has to complete a specific script. In Chapter 4, we argued that following a pre-set path can also be pleasurable, since it is similar to synchronizing to a pre-existing pattern. Completing an adventure game means restoring the behavior outlined by the game design.

Going back to our comparisons with theatre, the player of adventure games is supposed to “play the part,” but without having been given the script—finding out what

the game has in store is also part of the game. As in theatre, the player needs cues to know what to do next, such as making clear what the goals of the game are, and giving information to solve puzzles.

One of the most useful concepts to understand how cue-giving works in videogames is Murray's "scripting the interactor":

The lesson of *Zork* is that the first step in making an enticing narrative world is to script the interactor. By using literary and gaming conventions to constrain the player's behaviors to a dramatically appropriate but limited set of commands, the designer could focus their inventive powers on making the virtual world as responsive as possible to every possible combination of these commands. (*Hamlet on the Holodeck* 79).

Originally Murray is talking about (literary) conventions as a way to implicitly give cues to the player about how she is supposed to act and create expectations about how the simulation works. The example Murray uses is *Zork*, which has a large number of possible actions that can be used in it; the constraints established are conventions borrowed from *Dungeons and Dragons*, amongst other sources. The concept of "scripting the interactor" can extend further than to genre conventions. Montfort explains that the "script" that the interactor is supposed to follow is not like the text of the play, but is closer to the concept of "script" in Artificial Intelligence, which is the knowledge that is needed in order to respond to a specific situation. Montfort gives the classical example of the "restaurant script" (Shank and Abelson), which implies knowing when to sit down and where, how to order a meal and asking for the bill, amongst other things ("Fretting the Player Character" 140). Scripts are also part of the domain knowledge needed to solve a puzzle, since they are standardized sets of behaviors that the player should aim at reproducing. Scripting the interactor directly affects the player character, which can also



invoke a specific domain. Sherlock Holmes, who is the protagonist of sundry adventure games, invokes knowledge of Victorian England, which includes knowing about class differences, or scientific knowledge of the time. Holmes also has certain behavioral scripts associated with him, from being able to perform chemical tests (*The Lost Files of Sherlock Holmes: The Case of the Serrated Scalpel*), to being a master of disguise, and beating Inspector Lestrade at solving difficult cases.

Indirectly controlling the player's behavior is a compelling device in the design of adventure games (and videogames in general). By giving cues to the player, the simulation can communicate what to do without being explicit or overbearing, thus designing the experience while giving the player enough leeway to explore the world. Based on Murray, one basic way in which the game can script the interactor is by establishing what actions are possible in the world. In the case of adventure games, that means defining what the verbs of the game are. The verb menus in graphic adventure games, for example, are a way of constrain and design what the player character can do in the game. Therefore, the design of the player character shapes the participation of the player in the simulation.

#### 8.4.2.2. Design Elements That Define the Player Character

Scripting the interactor is therefore based on the design of the player character. By being able to reproduce the script, the player is able to restore the behavior of that character, and reproduce who that character is. The identity of the player character is another component of the process of restoration, since it helps the player narrow down the possible performances of the character. The following sections deal with the different identity markers and which ways they facilitate this type of “scripting.”

#### 8.4.2.2.1 *Image*

The first way to define a character, as Isbister and Meretzky (“Building Character”) argue, is through its external aspect. It is a fast way to create a character and establish expectations about who it may be; creating characters visually is something that the fine arts have been doing for many centuries. The representation does not have to be exclusively visual, though—many Infocom games included a description of the player character, which the player could read by typing “examine self” or “look at me.” However, the descriptions would be rather vague, given that the characters in text adventure games were usually undefined, as we will see below. In *Plundered Hearts*, for example, “look at me” is responded with “You are wearing a cotton frock, very pretty, if a tad outmoded for today's fashions” (what the player wears is important during gameplay).

The visual representation of the player character can give the player a lot of information at a glance: gender, age, attitude, profession, status, state of mind. The visual design of authors has been covered extensively by other authors (Adams and Rollings; Meretzky, “Building Character”; Isbister), so we refer the reader to those sources for extensive arguments on that topic.<sup>42</sup>

#### 8.4.2.2.2 *Name*

What’s in a name? Names are the most basic way to identify someone, distinguishing that person from others. A name can also invoke a domain knowledge, or a

---

<sup>42</sup> For a good overview of character creation based on visual design, see Will Eisner’s *Comics and Sequential Art*.

personality: King Graham (from the *King's Quest* series) invokes an Arturian-like setting, whereas Manuel “Manny” Calavera already has references to a noir-like world (the use of a short anglicized nickname for a Spanish name) and to the Día de los Muertos that inspires the looks of the game (“calavera” means “skull” in Spanish).

#### 8.4.2.2.3 Gender

There is a whole academic field dealing with how gender roles are culturally constructed, and how certain actions can be culturally associated with a gender or another. Therefore, defining the gender will generate certain expectations about what the player may be able to do (or not).

Many text adventure games do not even specify the gender of their protagonist. According to Atkinson, roughly half of the Infocom games do not specify the gender of the character (Atkinson, “Part I”). One of the exceptions, and the only Infocom game where the protagonist is female, is the text adventure game *Plundered Hearts*, where the heroine does not have a name, but we know she is a woman—the lady-in-distress of a romance novel, to be more precise. As such, there are certain actions that are beyond her realm of action, such as swashbuckling. The player must find ways to circumvent the limitations that are imposed by being a woman—for example, in order to be able to climb out a window, she must be wearing breeches, since climbing in a dress would be unseemly.

#### 8.4.2.2.4 Genre

As Murray noted when talking about the lessons of *Zork*, genre fiction is one of the basic ways in which to script the interactor; it can also invoke specific domain

knowledge in relation to the fictional world. The genre sets up the expectations of the player, as well as anticipating the actions that the player may have to do. Continuing with the example of *Plundered Hearts*, the game evokes the romance novel genre, set in the pirate-infested Caribbean of the late 17<sup>th</sup> century (two genres here, romance and pirate adventures). The player character faints at the beginning of the game, after her ship has been assaulted by pirates who are there to help her. It is not an action that the player chooses, she does not have control over it (fainting is not something people do voluntarily anyways), but it is in keeping with how ladies behave in romance novels with pirates. The player character will have to cross-dress, flirt with the villain who has kidnapped her father, and fight scary pirates (although she is not very good with the sword). In *Plundered Hearts*, the player character is not a proper romance novel heroine; if she was, she would have to be more passive, letting the male protagonist rescue her and do all the fighting. Passivity does not make for an interesting player character—the constraints on the player character set up interesting gameplay, but also offer the opportunity to overcome those constraints and therefore subvert the romance novel genre in the process.

#### 8.4.2.2.5 Actions

F. Scott Fitzgerald put it very succinctly: “Action is character.” Interactivity distinguishes videogames from other media as a participatory medium—the player has to perform so events take place. Thus the character, in this case the player character, is the fulcrum of player’s performance in adventure games.

In the current context, “action” can mean three different things in relation to the player character: the set of verbs available to command the player character, the actions

that the player character carries out without the player controlling it, and the actions that the player has to perform in order to restore the behavior that corresponds to that particular character.

The first and most evident device to use actions to define character is by selecting the verbs associated with it. In terms of game design, these verbs are the mechanics (Järvinen), the actions that the player will be able to perform repeatedly during gameplay. With respect to adventure games, the mechanics are the verbs that will be used most often (in the case of text-based input), the verbs that are part of the menu in menu-driven graphic adventure games, or the contextual mechanics that select the action depending on what entity the player interacts with.

Ben, the player character of *Full Throttle*, presents the possible actions in the form of five icons associated with different body parts: a fist (which corresponds to the verbs use, pick up or punch depending on the context), an eye (look/examine), a tongue (talk to or taste, again depending on context), and a boot-clad foot (kick). Ben is a biker, therefore counting punch and kick as two of the core actions establishes what kind of biker he is. In fact, the first action the player performs in the game is punching open the lid of the dumpster into which Ben has been thrown. The verb list does not include “open,” thus in order to open a bar door, Ben will have to kick it. The set of verbs is expanded by a mini-game, where Ben must ride his bike through meandering roads plagued with enemy cyclers. While riding, Ben must punch, kick, or use some sort of weapon (from a plank to a chainsaw) to defend himself from other bikers. All these actions profile the player character into a tough guy, and those are the actions that the player must perform (successfully) in order to make him come to life.

*Gabriel Knight: Sins of the Fathers* is another excellent example of defining the character through its actions. In the verb menu, “talk” and “ask” are two different icons, which would usually be conflated in most other adventure games. There is a subtle but important difference between “talk” and “ask” here. “Talk” will start off chit-chat with a character, but the player will have no choice about what to say; whereas “ask” will allow Gabriel to interrogate another character. This will bring up a set of dialogue options, which refer to different conversation topics. When Gabriel talks to a character, they may provide some general information about who they are or what they do; not all characters in the game are willing to be interrogated. “Ask” is a more useful verb, since it not only allows the player to choose what to say, but lets the player discover new topics of conversation to ask about. (The dialogue system will be dealt with in more detail in the following chapter.) The reason why they have different verbs goes beyond having a more detailed interaction for one particular type of actions: Gabriel is researching a book on voodoo, and as the game advances, he becomes a sleuth of sorts. Thus the game revolves around finding out information and using it to answer all the questions that pop up through the story. While “talk” will be similar to examining an object, “ask” is the core mechanic associated with Gabriel Knight. The conversations that take place as Gabriel “asks” other characters are also recorded; the tape recorder is another menu option where the player can have access to all the transcripts of the conversations.

The pre-scripted actions and the dialogue of Gabriel Knight are also excellent examples of giving a player personality through his actions. Whenever he is interrogating a female character, there will usually be an option to flirt with the woman he is talking to. If the player uses the verb “pick up” with a character, the game responds to it as if it was

“flirt” rather than “take” (the usual response is that the character in question is not his type). Finally, whenever Gabriel passes by a mirror, he automatically checks himself out in the mirror and fixes his hair—Gabriel is a ladies’ man, and concerned with looking good. This final detail is a representational device borrowed from dramatic media—a gesture that tells the player something about the character. It is also significant that some of these devices are things that the player does not directly control—a lot of the flirting happens in cut-scenes, the gesture of looking at the mirror is a canned animation. Gabriel is being himself, he does have a personality and does things that the player may not directly intend.

The personality of the character can also be expressed through the game design, so that the player has to think like the character in order to solve the puzzle. *Monkey Island 2: LeChuck’s Revenge* is a prime example of this. Guybrush is now a real pirate, who keeps boasting about how he defeated the ghost pirate LeChuck in the previous game. But in the end, he is not as brave or as tough as he presents himself; his most salient character trait as a pirate is that he is a cheater and can do very mean things to others for his own profit. Through the game, the player has to think as Guybrush in order to solve a good deal of the puzzles.

Several puzzles involve cheating, since Guybrush is not the big shot he pretends to be. In Chapter 6 we described the spitting contest, which involves moving back the marking stones of the best spitting in the competition. Guybrush also enters a drinking contest with an old pirate, but he cannot hold his liquor. Thus he has to substitute non-alcoholic grog for the grog he is given. There is also a fishing bet, where a fisherman dares Guybrush to capture a bigger fish than his—Guybrush does not sit down with a rod

and wait, but rather goes to the Governor's mansion and grabs the biggest fish in the pantry. In all these examples, the player has to think laterally in order to solve the puzzle—the solution does not come from thinking about how to win, but how to cheat.

Being mean to one's own advantage is another strategy the player must constantly follow in the game. Guybrush gets a cook fired from his job by putting a rat in the soup he is cooking, steals the monocle of a cartographer (without which he cannot continue his work), gets a pirate girl thrown into jail in order to escape a warrant, saws off the peg leg of a pirate, and traps an annoying coffin salesman in one of his own coffins. The restoration of Guybrush's behavior thus involves thinking like he would, and behaving like a selfish cheating pirate. This design strategy is unfortunately not very common in adventure games; *Monkey Island 2: LeChuck's Revenge* exemplifies how to encourage the player to think and act like the character would as part of puzzle-solving.

#### 8.4.2.2.6 Dialogue

How a character speaks can also define who the character is: accents can indicate where someone is from, the jargon they use tells us of their profession. The type of language used in a game can also situate it in time and genre: from technical / pseudo-scientific jargon (*A Mind Forever Voyaging*, *Space Quest* series), to pirate speak (*Plundered Hearts*, *Monkey Island* series). Defining a character through language is something that dramatic writing has tackled for centuries, so we will not go into detail here, since there are more extensive discussions elsewhere (see Lajos Egri).

What is more relevant to game design, and to establishing the relationship between player and player character, is how dialogue is an action in the game. "Talk to" is one of the basic verbs in adventure games, and the most obvious action to perform in



relation to an NPC. As Emily Short highlights in the context of Interactive Fiction, one of the problems of conversation is that it cannot be expressed in the same way as in real life. Ideally, the player should be able to type what she wants to say, expressing both content and tone, and the NPC should be able to understand both what the player said and react to the tone and attitude of the player. A sophisticated system that would be able to parse natural language as well as intent is still somewhat beyond the possibilities of the technology, although Short is one of the theorist-practitioners who is taking small steps to achieve that.

There are different systems in adventure games that design how the player character talks to other characters, and how to obtain information from them. This will be dealt with in the following chapter, since part of the design problem here is creating NPCs who respond in a believable, human-like way.

#### 8.4.2.3. Types of Player Character in Adventure Games

Based on the features that define identity listed above, there are three types of player character in adventure games: undefined, generic and defined. Each of these types are also generally associated with a type of interface. Depending on how each feature occurs, each type of player character establishes a different relationship between itself and the player character.

##### *8.4.2.3.1 Undefined*

The undefined player character is practically a “blank slate,” and no personality traits are ascribed to it. Who that character is has no relevance to the story of the game, and the character itself does not have a backstory that may give shape to its identity. The

character is a vehicle in the sense that Montfort uses, it is more a vessel that explores the world and interacts with it.

It is very difficult to find a completely undefined character, because there are many and subtle ways of defining the identity of a character without having to write its backstory, as we saw above. Wood refers to this type of character as “the amorphous hero,” a presumed Everyman where the player character can project her motives and emotions.

Undefined player characters also establish a precarious relationship with the fictional world. As Wood notes: “When PCs have no defined character, they do not feel like part of the setting. Their interactions with NPCs tend to be bland, and they become outlookers and outsiders.”

Undefined player characters can inhabit simulations with few or no NPCs, they are not participants in the events of the story, as is the case of *Myst*. Rather, their story deals with the reconstruction of the backstory, the exploration of the space, rather than having their own.

An undefined character does not have an image either, which makes it more common in text adventure games. Adams and Rollings refer to this type of player character as “non-specific,” citing text adventures as the main example precisely because they don’t have an image—*Adventure* and *Zork* are two of the most obvious examples. However, part of their argument is that text adventure games were written “as if the player *himself* inhabited the gameworld” (Adams and Rollings 152). As we have seen above, this affirmation is questionable given the detachment between the player character and the player. There are also text adventure games where the player is actually someone

specific, such as Bilbo or King Arthur, as we will see below. Undefined characters are not exclusive of text adventures either—as we mentioned above, the player character in *Myst* is implicit, since it is a first-person game, and not defined.

Undefined characters do not have a name or a gender, which are identity-defining features. Some text adventure games prompt the player for a name and gender (these are the only examples of configurable characters in adventure games); Atkinson (“Part I”) lists *Ballyhoo*, *Bureaucracy*, *Beyond Zork*, *Leather Goddesses of Phobos* and *Moonmist* in the case of Infocom. However, these markers of identity only define the name and gender that the NPCs will use to refer to the player character.

#### 8.4.2.3.2 *Generic*

As we have already discussed, the genre of the story of the game is a powerful device to define the identity of a character. Generic characters are relatively common in text adventures, since they can leave out identity markers, such as the image, name or (very often) gender.<sup>43</sup> The identity of a generic character is set within the conventions genre fiction, such as detective novels or science fiction. In *Deadline* the player character is a detective who must find out who killed Mr. Robner. In order to do so, the player character has to cross-question witnesses, examine the crime-scene, and gather evidence. Significantly, there is no reply to the “examine self” type of command, since “self” is not part of the game vocabulary, and “look at me” is replied with a default (and ungrammatical) message: “There’s nothing special about the me.” This lack of an

---

<sup>43</sup> A point-and-click adventure game using first person point of view has the potential to have generic characters. However, the author has been unable to find examples of non-text adventures with generic characters.

image again emphasizes that the story the player is concerned with is that of the crime scene, the Robner estate, and not the player character's.

#### 8.4.2.3.3 *Defined*

A defined player character has a name, which automatically identifies it as an individual, as opposed to the Everyman that the undefined character represented. The introduction of graphics seems to have been a cue for the generalized introduction of defined player characters even if the character was not depicted in the graphics. Graphical text adventures, such as *The Hobbit* or *Arthur: The Quest for Excalibur*, are two early examples of defined player characters.

The defined player character is the most extended type in adventure games. Generally, the player can see the player character in the simulation, moving around and acting in it. Their identity is thus defined primarily by their looks: Roger Wilco in the *Space Quest* series is a paunchy space janitor; Sherlock Holmes wears his perennial cape and hunter hat; George Stobbart in *Broken Sword: Shadow of the Templars* is a contemporary American tourist traveling around the world.

The player may identify with the defined character, similarly to how audiences can identify with the protagonist of a film. Defined characters, however, also tend to have a distinctive personality, which sets them apart from the player, again emphasizing the gap between character and player. The player is outside the simulation, participating and performing in it by controlling an entity that is actually part of it.

Some games highlight the difference between character and player. Defined characters can rebel when the player wants to have them do something stupid, or make a comment when it is obvious that the player is trying everything because she does not

know what to do. George Stobart in *Broken Sword: Shadow of the Templars*, for example, is often quite derisive whenever an action is not successful, or an action that the player tries does not make sense.

Other games point out the gap between the player and character, mostly to poke fun at it rather than to make an ontological statement. In *The Secret of Monkey Island*, Guybrush Threepwood pays to get swordfighting classes. After the classes are finished (the classes are represented in a cut-scene), it turns out that he still has a long way to master swordfighting, and that the learning is actually up to him. As Guybrush exits the house where he received his classes, he looks directly at the player and says: “I can’t help but feel like I’ve been ripped off. I’m sure you’re feeling something similar.”

*Day of the Tentacle*, the sequel to *Maniac Mansion*, throws a jibe at the player by pointing out at the gap between player and character. One of the puzzles of the game involves playing a practical joke on one of the characters, Ed. Afterward, Bernard (the player character) makes an apology of sorts: “Sometimes I do stupid stuff, and I don’t even know why... as if my body were being controlled by some demented, sadistic puppet master...” This joke also makes fun at some of the actions that the player character performs in order to solve puzzles; the restoration of behavior may make sense in the context of the simulated world, but not outside of it. In the case of *Day of the Tentacle*, these actions include putting a hamster in a freezer, defrosting it in a microwave<sup>44</sup> and dressing it up in with a conveniently shrunk sweater.

Defined player characters do not exclude the player completely, since they still

---

<sup>44</sup> This is a reference to something that the player characters could do in *Maniac Mansion*, which we saw in the previous chapter.

need the help of the player, so to speak, to overcome the obstacles in the game. For example, in *The Secret of Monkey Island*, the player character Guybrush is rather naïve, so it is up to the player to solve the puzzles. After obtaining what was announced as a map to the treasure of Mêlée Island, Guybrush realizes it looks like a diagram for dancing steps, and confesses to the player he has been swindled. As we explained in Chapter 6, the player is the one who has to figure out that those dancing steps are actual navigation instructions to walk around the forest, and guide Guybrush to the treasure according to those instructions.

Given the evolution that we have seen in the identity of the player character, we can affirm that, contrary to what Adams and Rollings state, the player character of adventure games is distinct from the player. Only as the identity markers become more specific is there less room for the player to fill out who the player character is. The restoration of behavior has to combine both the personality of the character and the actions of the player. Defined player characters have to be compelling enough, and react to the player's commands in engaging ways, because they are going to be the companion of the player through the game.

These different types show another historical trend of adventure games towards more defined characters. This definition seems to rely more on representation (e.g. image, pre-scripted actions), and restriction of the player's interaction (increased use of contextual mechanics), rather than in "scripting the interactor." This evidences how adventure games, in spite of being simulations, have evolved to restricting player's agency in order to facilitate the restoration of behavior, increasingly leaving more of the restoration to the performance of the computer.

### 8.4.3. Multiple Player Characters

Although so far we have considered adventure games to have one player character by default, some adventure games present more than one character that the player can control, one at a time. Depending on the game, the player can choose which character to use at any point (e.g. *Maniac Mansion* and its sequel *Day of the Tentacle*, *Suspended: A Cryogenic Nightmare*), or the game will dictate which character will be used depending on where the player is in the story (*The Hitchhiker's Guide To the Galaxy*, *Indigo Prophecy/Fahrenheit*). Multiple player characters can enrich the interaction of the game, since they increase the ways in which the player can perform in the world; at times, as in the case of *Maniac Mansion* or *Indigo Prophecy*, they can be used as devices to introduce replayability in the game.

When a game has multiple player characters, they have to be *defined* player characters. They must be distinct to each other, so there is a reason to divide the interaction with the simulation between different entities. Multiple player characters provide different points of view about the world, as Wood argues; depending on how they perceive the world, they will provide the player with different information. However, multiple player characters may also have distinctive abilities. As we saw above, actions also define character, so it makes sense that every character may have different verbs associated with them.

Having multiple player characters further evidences the detachment between the character and the player. A set of player characters is similar to a character party in a CRPG (e.g. *Final Fantasy* series, *Planescape: Torment*), where the abilities of every character often complement each other. In adventure games, having multiple characters

usually means that the game cannot be completed by controlling one character alone, but by working with them in combination. The player must choose who to control in order to obtain all the necessary knowledge to solve the puzzle, and then control the character with the abilities that will help tackling the problem at hand.

Thus multiple characters can somewhat complicate the restoration of behavior, particularly in the cases where the player has to choose between several characters. The performance of the player has to split up in several realms of influence, which are associated with each character. The player must choose the right character at the right time, making it easier to lose track of what the behavior to be restored should be. There is a recurring shift on the point of view within the fictional world, which can be confusing to the player if not managed consistently, as we will see below.

It is somewhat rare to find multiple player characters in text adventure games, firstly because of the abundance of undefined / generic characters. The lack of visual feedback is also a vital factor for not having several player characters, since it is difficult for the player to know who she is controlling at each turn without having continuous visual feedback. However, there are a couple of interesting examples of text adventures with multiple player characters. *Suspended: A Cryogenic Nightmare* presents a player character who is in suspended animation stowed away in an underground complex. The player character controls all the vital systems of a planet (e.g. weather, food, transportation) while something has gone awry, so that the player must stabilize the systems before he gets disconnected from the machine that hosts him. Rather than controlling the “suspended” player character, the player gives orders to six robots in the underground facility. Each robot has specific technical abilities, which allow them to



provide information about the world differently: Iris can obtain visual information only, Whiz can check up information from the central computer, Waldo can grab objects and move them physically, Auda can receive audio signals, Poet can diagnose electronic systems and communicate in mystifying terms, and Sensa can detect magnetic and photon emissions. Since their ways of acting and perceiving the world are complementary, the player must combine the abilities and perceptions of all the robots in order to try to fix the planet's system. In order to facilitate tracking where each robot is, the game box included a map of the facility and tokens representing each robot—the game is quite complex, and needs extradiegetic visuals in order to be played.

Another example of a game using multiple player characters is *The Hitchhiker's Guide to the Galaxy*. The player controls Arthur Dent during most of the game, but there are certain sections where the control will switch to other characters, such as Ford Prefect, Zaphod Beeblebrox, or Trillian. The game switches the player character automatically, depending on where the player is in the story, saving the pains of having to track who the player controls each time.

Multiple player characters are easier to implement in graphic adventures, where the player can see what character she is controlling. *Maniac Mansion* is a good example of how different abilities can also define the personality of a character. The premise of the game is that a cheerleader, Sandy, is being kept hostage at Maniac Mansion, and her friends are going to rescue her. Dave Miller, the victim's boyfriend, heads the party—he is personally invested, and he provides continuity to the ideal walkthrough. The player has to choose two other characters, out of six total, to go with Dave; each character has different attitudes and abilities. Having multiple characters introduces replayability in the

game, since there are different ways of traversing the game depending on whom the player chooses at the beginning.

The personality of each player character in *Maniac Mansion* is defined by what they do. For example, Bernard, the geek of the group, is the only one who can fix electronics; he is also afraid of a certain Green Tentacle and will not go near it, preventing him from accessing certain parts of the mansion. Wendy is an aspiring writer, which enables her to re-write documents. Michael is an amateur photographer and knows how to use the darkroom in the house. Some of the characters are sloppy, and will make a mistake in one of the puzzles if the player chooses them because they cannot perform the action correctly (they'll give a package away without retrieving the stamps first), which will prevent the player from finishing the game. Whom the player chooses determines what puzzles will be needed to complete the game, which means that the designers had to anticipate all the possible solutions. *Maniac Mansion* is a rare case of a multiform story in adventure games (Murray, *Hamlet on the Holodeck* 30-8), where some sections of the story will feature different events depending on the choices of the player, thus encouraging replay. In this case, designing different characters also means designing different solutions for some puzzles.

The sequel to *Maniac Mansion*, *Day of the Tentacle*, did away with the problem of having to design different solutions to a puzzle, because the three player characters were pre-determined (Bernard was again one of them). What the player can do is choose which character to control at each time. The player characters have distinct personalities, which are reflected in the writing and their attitude towards the fictional world, but not different abilities. Rather, their realm of influence is separate: each character has ended

up in different periods of time in the same location: one goes 200 years into the past, when the U.S. Constitution is being written, another stayed in the present, and another was sent to the future, when tentacles rule the earth. The player can send objects (but not living beings) from one period to another using a time machine, which works as a connection between realms. Thus the puzzles consist on figuring out which period of time to use an object, also taking advantage of the passage of time to change some of the objects.

*Indigo Prophecy* / *Fahrenheit* also features multiple characters, where each character is supposed to provide a different perspective on the story as it unfolds. *Indigo Prophecy* is another example of a multiform story integrated in the game, based on choosing certain characters at certain times. The variation in the story is not significant;<sup>45</sup> most times the change just refers to which character carries out certain actions.

The game is divided into chapters; in most of the chapters, the player is offered the possibility of choosing which character she wants to control. This often means choosing in which order certain parts of the game will be completed; only in a few instances will the player go through a specific chapter using one character instead of another. The different points of view provided by each of the player characters seek to engage the player in having to put together all the pieces of information—the game is a murder mystery at heart. The goal of the player is to obtain as much information as possible through the characters, whereas each individual character only knows part of what has happened.

---

<sup>45</sup> The game has five different endings, which are different cut-scenes depending on what the player chooses in the last chapter and how well she does in the final fight.

*Indigo Prophecy / Fahrenheit* also exemplifies some of the issues that crop up when designing an adventure game with multiple player characters. The player controls four characters through the game. Lucas Kane is the character the player controls the longest; he has murdered a man while in a trance and does not remember anything. His goal as a character is to find out what happened to him, and prove that he was not in control of his actions while he committed the crime. The player controls three other characters throughout the game: Carla Valenti and Tyler Myles, the detectives investigating the homicide that Lucas was involved in, and Lucas' brother Marcus Kane.

The goals of player characters in *Indigo Prophecy/Fahrenheit* are in conflict—while Lucas wants to prove his innocence and not end in jail, the two detectives are trying to hunt down Lucas. These conflicting goals get in the way of who the player identifies with, in contrast with the games listed above, where all the characters have a common goal. This conflict is evident in chapters where two player characters meet, but only one of them is controlled by the player. In one of the segments of the game, the player controls Lucas as he is being cross-questioned by one of the detectives (which detective is in the scene depends on previous choices of the player). The player can choose to be truthful, avoid the questions, or lie. Since the player is also aware of what the detective knows, there is a contradiction between whose goals she should pursue; which character is the focus of the performance is confusing. This choice becomes even more complex when immediately after the cross-questioning, the player controls the detective and has to find the evidence that she had previously hidden while controlling Lucas. In the end, it does not matter who the player wants to side with—if Lucas gets caught, it is game over and the story ends there. The proper behavior to be restored has to

be both providing the detectives with evidence while allowing Lucas to escape. The player has to go through a set of different actions that allow advancing the story, but which are divorced from the goals associated with each different character. If the player tries to do something that is not part of the established script, she will fail.

The shift between player characters in *Indigo Prophecy/Fahrenheit* also displays some problematic design decisions, particularly when it comes to dialogue. The player starts controlling the character she has chosen, and the conversation starts. The conversation system displays different topics that the player can talk/ask about, summarized in one word; the player has to choose one within a short time span, as the system is supposed to mimic real conversation. The problem is that, once too often, the game switches whose responses the player is choosing in the middle of a conversation. This happens, for example, the first time Lucas meets his brother Marcus in the game—the focus shifts back and forth between both characters, without giving cues to the player that establish whose dialogue she is selecting. It is confusing, and seems in conflict with the idea of mimicking real conversation that the timed responses seemed to aim at—in real life, we only control one side of the conversation.

These two instances from *Indigo Prophecy/Fahrenheit* evidence that the relationship between the player and the player character is different from other adventure games. The game seems to conceive the role of the player as closer to a “movie director”, thus the expected performance is that of someone who orchestrates what happens in the fictional world—the introduction to the game, with a digital movie set, seems to imply as much. The player cannot focus on one character at a time, but will rather jump from one to another in order to restore the behavior implied by the game. *Indigo Prophecy /*

*Fahrenheit* also evidences that, in spite of the detachment between player and character in adventure games, the player needs to anchor her actions with one character at a time, so that the character works as the control locus of her interactions within the simulated world. Not knowing whose dialogue is being chosen is jarring, and can also make the player feel she is not in control of her performance, rather than being a movie director. Losing the control locus means severing the connection between player and the character; the designer therefore takes away the power of decision from the player. As we have argued above, adventure games are not based on choice, since the goal is to restore a set of behaviors. However, one of the premises of the game is that player's choice affects the story, as stated both in the prologue and in promotional interviews (see MacDonald). Not only is that not true as the game advances, but in the long run player's choice in *Indigo Prophecy/Fahrenheit* becomes even more restricted than in many other adventure games, since there is still an embedded story that the player must re-enact.

#### **8.4.4. Death of the Player Character in Adventure Games**

There are two schools of design in adventure games. In the first one where the player character can die, or at least reach a "game over" state before the intended end of the game; in the other the player character cannot die, i.e. the only game over state is the end of the game. These two approaches have diverging implications, which affect the overall design of the game, and designates how the player performs in the game. The "mortality" of the player character determines how the player learns how to restore behavior in the game.

#### 8.4.4.1. The Player Character Can Die

Early adventure games let the player character die if the player makes a mistake, or loses a fight. It is “game over” as it was in arcade games, where you had to start the game from the beginning if you lost. *Adventure* and *Zork* incorporate combat, inspired by table-top role-playing games; if player character is defeated, it is the end of the game, and the player has to re-start. In later games, the player has the option to save and restore the game; however, Crowther’s version of *Adventure* does not include any commands to save the game (Jerz), turning the death of the player character into an important loss.

Since there is a fair chance the player character may meet an untimely end, the player is kept on her toes the whole time. The restoration of behavior in this type of game expects the player to save her game often, thus having her resort to extradiegetic devices to be able to advance in the game. The decisions the player makes are very important, since a bad choice could have fatal consequences (e.g. entering a dark place without a lantern can get the player eaten by a grue in *Zork*).

Killing the player character also informs the player immediately that she made a mistake, becoming a brief and somewhat traumatic experience. This clear feedback tells the player that the behavior she tried to restore was not proper. Death does not necessarily mean that the player character is deceased; it can also be any state of affairs that will prevent the player character from getting to the end of the game. For example, in *King’s Quest V*, King Graham can be turned into a frog, which prevents him from rescuing his family from the claws of an evil wizard. In *Indigo Prophecy/Fahrenheit*, if Lucas is caught by the police, it is the end of the story. Interrupting the player’s performance because she did something wrong can also undermine the simulation—exploring the

system is still necessary, but it can be punished more often than rewarded.

The embedded story in adventure games often dictates that there are certain things the player will be prevented from doing by the design of the game—the restoration of behavior can only happen in a very specific way. Rather than telling the player that it is not possible to do that, the player is directly banned from doing it. Killing the player character is a clear message that something cannot be done, or will not help the player complete the game. However, it must be clear to the player why the player character died, and what her mistake was. Dispatching the player character as a way to get the player do what the designer wants her to do, without giving substantial reason, is frustrating to the player, and also may denote a lack of resources on the part of the designer. As we saw above, there is a range of devices to “script the interactor,” that do not require the player to make a fatal mistake in order to know what she is supposed to do.

Having the player character die in order to learn the solution of a puzzle is a design problem. As Graham Nelson points out (“The Craft of Adventure”), the player should have all the information necessary so that she is able to solve a puzzle without trial and error. When trial and error involves the player character dying, it rattles the foundations of the logic of the fictional world—the player is forced to step out of the game and bring the experience of “previous lives” as Nelson calls it, in order to solve a puzzle. In the particular case of enforcing death, the game is not providing all the necessary information about the fictional world, which will not allow the player to solve the puzzles in the simulation.

The death of the player character, therefore, extends gameplay beyond the fiction of the game into the extradiegetic realm—the player is the one who learns the lesson, not



the character. In a way, it increases the breach between player and player character. For example, in some of the earlier *King's Quest* games, when the player character died, a “thank you for playing” message appeared. The message referred to the player as “you,” who had made a mistake, and made a reference to “we” as the game makers at the company. In *King's Quest IV: The Perils of Rosella*, one of the death messages included a picture of Roberta Williams, designer of the game, with a message for the player. “Thank you for Playing King's Quest IV, ‘The Perils of Rosella.’ Next time... be more careful!”

Some games turn dying into an art, coming up not only with bizarre ways to die, but also with funny messages communicating it. *The Hitchhiker's Guide to the Galaxy* “rewarded” the death of the player with a Douglas Adams-esque description of the death of the character. Later on, the *Space Quest* series made a trademark out of having Roger Wilco dying bizarre deaths, and mocking the player for her mistake.<sup>46</sup>

The humor in both games certainly favors the jokes about the player character's death. In the case of *Hitchhiker*, the game is rather difficult, so the player character will die repeatedly. Even though the descriptions are fun, in the long run they can also be quite frustrating, since they not only interrupt gameplay, but also the player can lose a good deal of her progress in the game if she does not save often.

#### 8.4.4.2. The Player Character Cannot Die

The other side of the coin is not letting the player character die. This design

---

<sup>46</sup> One of the walkthroughs of *Space Quest: The Sarien Encounter* consists of a complete list of all the possible deaths in the game and when they appear: <http://www.gamefaqs.com/computer/doswin/file/562689/19842>

choice has become more generalized in adventure games, since the Lucasfilm/Lucasarts games popularized it in the late 80s. Not having the player character die can prevent the frustration caused by dying repeatedly in a game, and focuses the player on exploring the simulated world. If the player makes a mistake, it is always in-world, and has no deadly consequences; there is no risk of the player losing her progress of the game because she failed to save the game before making that mistake.

The absence of “game over” states characterizes simulations that are more open, where the player can wander around and explore and solve the puzzles at her own pace. However, wandering can also become aimless when it goes on for too long and all that there is to explore has been exhausted. The player can get stuck and not make any progress because she does not know what to do next, but there is no direct feedback telling her what she is missing. Death messages tell the player clearly what she did wrong, and prompt her to correct it in the next replay. With no death messages, it is more difficult to know what the goal of the player’s performance is. This requires the goals of the game, both short and long term, to be clearly stated to the player in the story, and always give enough information to the player to know what to do next.

Not letting the player die encourages exploration, because the player is aware that she can examine every nook and cranny available without running the risk of losing her progress. However, this can also lead to indiscriminate trial-and-error in solving the puzzles, rather than trying to figure out the solution from the information available to the player. The feeling of security can also encourage the player to pay less attention to what she does in the game. There are no wrong decisions; in fact, there are no decisions at all—the player is allowed to roam around the simulation until a certain state of affairs is

reached. Eliminating death means that the player is not penalized for her mistakes, but it also removes the sense of consequence for having chosen to do one thing instead of another.

### **8.5.Conclusion**

The player character is a key design element in adventure games. It is more than a bridge between the player and the game—it provides the player with a point of view in the fictional world and with agency in the simulation. We have stayed away from discussions of how to build character in the representation, because there are plenty of other authors that have dealt with it before—it is one aspect where videogames can learn from other media, such as theatre, comics or film.

The goal here has been discussing how game design helps in guiding the player to complete the story of the game. In the case of adventure games, players have little leeway to create their own characters, partly due to the need to complete the embedded story, partly because higher levels of abstraction reduce the number of available actions. Playing the game does not mean playing the character, or finding an identity in the simulated world—it is a process of restoring the behavior associated with that character.

The player character does not confront an empty simulated world, although games like *Myst* may be close to it. The simulated world is populated by computer-controlled NPCs, whose role is not only making the world lively, but also become part of the successive states of affairs in the game. That will be the topic of the following chapter.

## **CHAPTER 9**

### **NON-PLAYER CHARACTERS**

#### **9.1. Populating the Simulation**

Non-Player Characters (NPCs) are both existents in the fictional world and the agents that are controlled by the game itself; they are co-performers in the simulation. They make the space come to life, and turn the simulation into a social environment. NPCs also take part in the puzzles of the game: they can be part of the state of affairs, assistants in the puzzle solving, or the opponents to the player characters. Some aspects of the NPCs are represented, as was the case of the player character, such as how it looks and speaks. However, their design is more problematic, since their simulated behavior and dialogue have to give the appearance of life. While player control was the spark to life of the player character, the challenge of computer-controlled NPCs is to simulate the human-like behavior.

The actions and reactions of NPCs are part of the rules of the game, particularly the rules of the simulation that are implementing the fictional world. The key difference between objects and NPCs in the simulation is that the latter are animated, i.e. they have the appearance of life. “Animate” has its etymological origin in the Latin term “anima”, which means “soul.” Therefore the term animated, in this context, should be what looks like it has a human soul (even if they are animals or machines). How can we program the appearance of a soul?

The simulation of NPCs is thus problematic—recreating believable behavior as a rule system is quite difficult. Although the field of Artificial Intelligence has gone a long

way to create agents that may appear intelligent, simulating characters that respond to input as a human would still constitutes a challenge.

The story-driven nature of adventure games has made them the earliest videogame genre to deal with the problem of NPCs. However, as the simulation of the game has become less and less nuanced, adventure games have also become the first genre to have given up on designing more sophisticated NPCs, leaving the personality of the character almost exclusively dependent on dialogue writing. The representation, rather than the simulation, takes the responsibility of animating the character, letting traditional dramatic and fictional writing techniques substitute for an artificial character performing in accordance to the rules of the game.

The following sections survey the functions of NPCs in adventure games, the strategies that the genre has used to bring NPCs to life, and the limitations that the design of NPCs has run into.

#### **9.1.1. Simulating the Characters**

Adventure game NPCs are often not so animated as to give the illusion of life. As Aarseth argues:

The greatest aesthetic problem for the adventure story-game seems to be believable characters. In *Adventure* there were just animals and monsters, and hardly any dialogue, and in *Myst* there were no characters at all, except for in a few static video sequences (“Genre Trouble” 51).

Even though Aarseth is right in highlighting the problems to create believable characters, such as being able to have a conversation, or respond adequately to the player’s actions, this is not an issue exclusive of adventure games. Games that brand

themselves simulations, such as *The Sims* or *Half-Life 2*, also have problems creating believable characters. Videogames not only tend to be populated by caricatures with no psychological depth, their limitations as intelligent entities are constantly challenged by things as trivial as finding their way in the simulated world (pathfinding), or giving appropriate support in case the player is in trouble e.g. providing cover to the player during an ambush.

Although adventure games present a few exceptions to Aarseth's statement, it is true that believable adventure game NPCs are very scarce. It does not take the player long to find the limits of the pseudo-human behavior of the character: at times they seem to be stuck in a specific area of the world, or do not react to player interaction adequately, or get stuck in a loop and always give the same responses. The shortcomings of NPCs can be rendered evident in story-driven games, since characters are part of the existents of the story, and thus one of the basic elements of story.

An easy way to avoid unbelievable characters is by not having NPCs at all, either by leaving them out of the fictional world, or representing them in the story. Emily Short notes that NPCs can be characterized in absentia, i.e. through the memories of the player character, through evidence of the NPC's presence in the world (e.g. letters, personal effects), or by having other NPCs talk about that particular character. The represented NPC is not part of the simulation, but a piece of information. This is how the NPCs in *Myst* are part of the story—the player does not interact with them directly, but rather with the objects that they've left behind. Even when they finally appear, Atrus and his two sons, Sirrus and Achenar, appear as are video recordings with which the player cannot interact.

## 9.2.NPC Functionality in the Simulation

As in the case of the player character, the functions of NPCs are not mutually exclusive. Nevertheless, they frequently have one single function in the game, either as being part of a puzzle or as provider of information. The limited co-functionality results in less complex characters; using a higher level of abstraction makes them easier to design, but also makes them less animated.

NPCs have three basic types of functionality in the simulation: providers of information, part of a puzzle, or sidekick of the player character. Most of the interactions with NPCs take place through conversation, which can be the way to obtain information from them, or get them to do something.

### 9.2.1. Providers of Information: Conversation Systems

One of the main functions of NPCs is to provide the player with information about the fictional world and the goals of the game. NPCs give the fictional world depth and complexity, as well as contributing to its pre-history—they can tell the player character what has happened in the fictional world before the game started. As we saw in previous chapters, learning about the fictional world is fundamental in order to help the player restore the behavior, since that information will help the player solve the puzzles. For example, the pirates in the Scumm Bar in *The Secret of Monkey Island* explain that the ingredients of everybody's favorite drink, grog, are very corrosive, so the bar owner has to keep replacing the mugs where it is served. This information is crucial for a later puzzle—in order to transport grog from one place to another, it needs to be poured to a new mug as the grog dissolves the old one.

The information about the fictional world is usually obtained through

conversations with the NPC, so that most times the player character dialogue consists of a series of inquiries. Many adventure games allow the player to revisit conversations, i.e. go back and ask the same questions again. The NPCs will usually respond in exactly the same way, or repeat the same information, independently of the number of times the player character has asked. In the best of cases, there may be some acknowledgement that the information has been given before. For example, in *The Secret of Monkey Island*, Guybrush can ask pirates to repeat their insults during insult fighting: “Uuuh... Could you repeat that? I didn’t quite get it,” to which the pirate will respond with, “I said: [...]”

The possibility of revisiting information usually occurs in games where the player character does not die, or there is no time limit (this will be dealt with in the following chapter). These games encourage the player to explore the world exhaustively, and dialogue becomes one of the sources of information that the player can go back to.

Revisiting information, plus the fact that NPCs may not “remember” previous conversations, nor show annoyance at being asked about the same things repeatedly, affects the believability of the character. However, this may also be a design choice. Games like *Indigo Prophecy / Fahrenheit* or *Hotel Dusk: Room 215* have dialogue systems where the player only has one chance to ask about specific topics; if she misses that opportunity, she cannot go back and ask. This choice can be rather risky design-wise—if the piece of information missed is key to finishing the game, she may be stuck without knowing why. It is also difficult to point out to the player that she missed that information without actually giving it away; it is probably easier to be redundant. Thus the information missed is usually not essential, but rather extra information that allows the player to know more about the fictional world, filling the gaps of what is needed to



complete the game. Both *Indigo Prophecy / Fahrenheit* and *Hotel Dusk: Room 215* have episodic structures, so if essential information is needed, the player can go to the beginning of the chapter and try again. It is no coincidence either that they are also detective stories, where obtaining information about past events is the key activity for puzzle-solving. The more information the player has, the easier it will be for her to solve the case, even if the player character itself does not have all the information.

Adventure games have come up with different abstractions of conversations which have led to diverse design strategies. Each conversation system uses different levels of abstraction in order to facilitate the communication between the player character and the NPCs. The following systems go from the more nuanced to the least interactive system: natural language parsing, choose your line, topic-driven conversation, and canned conversation.

#### 9.2.1.1. Natural Language Parsing

This type of conversation system is exclusive of games that take text input, such as text adventures. This system is based on text parsing, which can follow two different models—what Roberts refers to as “cheap AI.” One type is keyword matching, which looks for keywords that might be meaningful, and gives a response according to that keyword, ignoring the structure of the sentence. The other type is “regular expression” matching, which uses the sentence structure as a way to “understand” the input. This is the system inspired by Weizenbaum’s *Eliza*, where the system took the sentence and turned it into a question in order to keep the interactor typing. These systems use direct style, i.e. the player types what she wants the player character to say.

The conversation systems in Interactive Fiction usually follow indirect style

(“Ask NPC about X”). Some Infocom games also allowed players to talk to the NPCs directly (e.g. “NPC, tell me about X”) as long as the name of the character opened the command. The following examples come from *Deadline*, which has one of the most accomplished dialogue systems in Interactive Fiction, and allows both formats of interaction. The first character we meet in the game is Mrs. Robner, the wife of the victim.

> mrs robner, hello

Mrs. Robner says a curt “Hello”.

> mrs robner, where is mrs robner

“Ahem...”

> mrs robner, tell me about mr robner

“I loved my husband, no matter what you may think. I am very sorry to have lost him.” The speech is almost a set piece, and not too convincing.

> ask mrs robner about mrs rourke

Mrs. Robner looks briefly toward Mrs. Rourke and then speaks in a low whisper.

“She’s an excellent housekeeper and sometimes even helps with the cooking. For all her good qualities, though, she’s too much of a snoop and a gossip. She tries to keep track of everyone around here. It’s unnerving sometimes. But, as servants go these days, she’s quite extraordinary.”

There are several remarkable features in these interactions with Mrs. Robner. The first and most remarkable is that the player directly types what the player character says, letting the player take over the character during the dialogue. However, the player can still use indirect style to keep some distance from the action if she so wishes—the player can choose how to play the character, even if it is in a very basic way. Also, instead of giving an error message when we ask Mrs. Robner where she is, the character responds directly to the mistake, implying that the sentence does not make sense—the error message is thus diegetic.

The player does not only ask about information, she can tell a character to do something, or talk about a certain topic. In the end, the keyword-based parsing turns the conversation into choosing a topic to talk about (“tell mrs rourke about mrs robner,” “ask mrs robner about murder”); the difference lies in how the player constructs the command, and how the more fine-grained interaction introduces a minimal level of identification with the character.

#### 9.2.1.2. Choose Your Line

When the player initiates the conversation, by selecting or typing “talk to character,” a list of possible dialogue lines appears. The player can then read what the PC will say before she selects it. This type of dialogue is also referred to as “menu-based” (Short; Roberts); it relates to the topic-driven conversation described below, because it uses a branching structure. Some dialogue options will lead to other options, because they are follow-up questions or issues related to what the player character just asked. This system usually requires the player to revisit the conversation until all the branches have been explored, since the options usually do not change. Certain options in the dialogue may appear after specific events in the game, e.g. having solved a puzzle, inviting the player to go through the dialogue again.

One of the best uses of the “choose your line” system is the insult-fighting in *The Secret of Monkey Island*. The key to sword-fighting is saying the right insult to your opponent to catch them off-guard. Conversely, the enemy’s insults can be deflected by having the appropriate comeback line. The player character, Guybrush, starts by learning two insults, and their corresponding comebacks:

Insult: You fight like a dairy farmer.

Comeback: How appropriate. You fight like a cow.

Insult: Soon you'll be wearing my sword like a shish kebab!

Comeback: First you better stop waving it like a feather duster.

The player must fight other pirates in order to “learn” more insults and comebacks. Whenever a pirate insults Guybrush in a new way, the line is added to the repertoire. Same goes with the retorts: whenever Guybrush's insults are foiled by a good, new comeback, the line is included in the responses menu. Insult fighting requires the player to fight and lose many times in order to obtain all the lines of dialogue. The final goal is to defeat the Sword Master of Melee Island, who uses new insults that can still be countered with the learned comebacks. For example, the first two comebacks that Guybrush learns can be used to fend off the following insults:

Sword Master's Insult: I will milk every drop of blood from your body!

Comeback: How appropriate. You fight like a cow.

Sword Master's Insult: My tongue is sharper than any sword.

Comeback: First you better stop waving it like a feather duster.

The insult-fighting of *The Secret of Monkey Island* is less a conversation system and more a simulation of how the player character learns. It uses a common device in menu-driven conversation: the menu options expand as the player character learns more information. In this case, rather than opening up branches of conversation, insult-fighting becomes a combat system based on dialogue.

#### 9.2.1.3. Topic-driven conversation

Topic-driven conversation is similar to “choose your line,” where starting the conversation also triggers off a series of menu options. The difference is that instead of

offering the line that the player character will say, the options will be different topics that the player character will talk about. These topics can be summarized in one word or a phrase (*Gabriel Knight: Sins of the Fathers*, *Indigo Prophecy / Fahrenheit*), or with an icon (*Broken Sword: Shadow of the Templars*). As we already noted, with text interfaces, this type of conversation takes the form “Ask Character about X,” resorting to indirect style, and removing the conversation one level in the interaction with the NPC.

Although this system offers the possibility to choose what to talk about, the agency of the player is somewhat curbed, since the player does not exactly know what the player character is going to say. This was also the source of confusion in the case of *Indigo Prophecy / Fahrenheit*, where the player cannot be sure who is going to speak.

Conversation systems based on topics tend to branch out less frequently than “choose your line.” That means they usually do not have a conversation tree; rather, they list the topics the player character can talk about.

In the previous chapter we introduced how *Gabriel Knight: Sins of the Fathers* had two different verbs to talk to other characters: “talk” and “ask.” Each verb corresponds to two different conversation systems. “Talk” starts off a canned conversation (see below), while “ask” is a topic-driven conversation system. Gabriel’s “ask” menu has two sections: the top section includes topics that all characters can be asked about, while the bottom options are based on the context of the conversation. Gabriel can talk to a character first, which will reveal different pieces of information, and then ask the same character about them.

The “ask” conversation system in *Gabriel Knight: Sins of the Fathers* also models learning by adding conversation topics to the menu. These new topics can be added by

talking to characters, examining the environment, or after something happens in the story (e.g. a new murder is discovered). The conversation topics that can be used with all the characters most times do not provide Gabriel with useful information. The list grows to fifteen topics by the end of the game, plus asking the character about himself/herself, so the menu also becomes a bit unwieldy. This system provides the player with many choices, and it takes a lot of questioning (and patience) to gain information. It is up to the player to use selective encoding in order to identify what may be a clue from what is not. In the end, the game does a good job of turning the player into a detective, by providing large amounts of information to wade through.

#### 9.2.1.4. Canned Conversations

This model presents a conversation that only the player can initiate; the dialogue is represented rather than simulated. When the player starts a conversation with “talk to” (either in graphic adventures or in text), a pre-scripted conversation between the player character and the NPC plays out. The player has no say about what information she would like to talk or ask about. This model was, for example, typical in the *King’s Quest* series. In this type of conversation, the NPCs become a “signpost” of sorts, where the player just reads the information that the character has to give, so the player has no agency over the conversation other than initiating it.

This type of conversation is the one that allows dramatic writing to animate the NPC, since the character is being represented rather than simulated. This results in reducing the simulation of the game, giving the impression that the player is reading an interactive novel or watching an interactive movie, rather than playing a game.

### 9.2.2. Part of a Puzzle

Since NPCs also have behaviors that are established by the rules, they can also be participants and agents in a puzzle, usually because the player can get them to do something. Providing information may or may not figure into making them part of the puzzle, depending on what information is given and how. For example, the important-looking pirates in *The Secret of Monkey Island* tell Guybrush what he has to do to become a pirate (master the sword, the art of thievery, and find treasure), but they do not help him achieve it. In fact, it is not necessary to talk to them in order to complete the game once you know the goals, so they are not part of the puzzle. On the other hand, training with Captain Smirk is one of the tasks needed to master the sword, although his training is not really the key to the mastery. Smirk is the one to point out that Guybrush has to learn the proper pirate insults and comebacks in order to defeat his opponents, so the insult-fighting mechanic is activated after talking to him. Only talking with Smirk will allow Guybrush to start insult-fighting, making this NPC part of the puzzle.

NPCs can also give information to solve a puzzle indirectly, rather than stating it in the dialogue. Continuing with *The Secret of Monkey Island*, Guybrush has to find where the Swordmaster lives in order to defeat her in a fight. The only one who knows how to get to her house is the shopkeeper. Guybrush must ask the shopkeeper to go to the Swordmaster's house and request a challenge with her. She invariably declines the challenge, so Guybrush must find her—he has to follow the shopkeeper through the forest in order to find the Swordmaster's house.

Characters can be part of a puzzle more directly by being either givers or receivers of objects. This way, NPCs relate to the “helper” in Propp's *dramatis personae*

of the folktale, i.e. the character who helps the hero to complete his quest, usually by giving him an object. When the NPC is on the receiving end, it is usually to provide the player character with something else (e.g. another object), or enable the player character to advance further in the game. Object-giving puzzles are basically lock-and-key puzzles, i.e. using one object to “open up” access to a new part of the game. Another example from *The Secret of Monkey Island*: the red-herring puzzle described in Chapter 6 requires the player to give a red-herring to the guard of the bridge, in order to be able to cross to the other side.

NPCs can also be part of a puzzle by actively opposing the player character and getting in its way. Functionally, this is the equivalent of the Proppian villain. The clearest examples of this type of adversary NPC are usually related to role-playing or fighting mechanics. For example, the Pirate in *Colossal Cave/Adventure*, or the thief in *Zork* are entities who roam around the world, and will steal all the player character’s items if they meet. In *Indiana Jones and the Last Crusade: The Graphic Adventure*, the eponymous hero has to fight his way through the corridors of the castle Brunwald to find his father whenever a Nazi soldier identifies him as an intruder. If Indiana Jones is defeated, the player has to start over (or restore a saved game).

NPC adversaries are a particular type of challenge—they are active agents trying to prevent the player from reaching a specific goal. As we saw in Chapter 4, puzzles are not “active” by nature, since the challenge is to reach a specific state of affairs, and usually do not involve other active agents. The pirate, the thief and the Nazi soldiers above move around the world, and will actively seek to stop the player character from finishing the game. Adversaries are thus relatively scarce in adventure games, since they



bring in a change of pace and shift the focus from puzzle-solving to combat.

### 9.2.3. Sidekick

Some NPCs work as assistants to the player character, again, in the role of “helper” as defined by Propp. They usually follow the player character through certain stretches of the game. In their quintessential form, sidekicks are never controlled by the player—their role is to assist the player throughout the game. The assistance can range from giving extra information to the player (e.g. hints), to helping the player solve a puzzle, being an independent agent, or even solving puzzles for the player. Sidekicks are supposed to be rather sophisticated NPCs, because they usually spend relatively long periods of time with the player character, and must react to the player’s actions in a believable, animated way.

The wild bunny Max in *Sam & Max Hit the Road* is a good example of sidekick, because he accompanies Sam, the player character, through the game, and performs certain tasks at Sam’s behest. However, instead of telling Max what to do, Sam “uses” Max with different objects and other characters—Max is actually part of the inventory. This objectification of Max is actually funny because, although it is in fact a character, it is wild and uncontrollable; “using” Max means unleashing him to do something unexpected.

Significantly, in the recent *Sam & Max* episodic series, although Max is accompanying Sam throughout the game, he also functions as a provider of information. He cannot be “used,” but rather the player can cue him to do something through dialogue and depending on the context. This mechanic is the result of the absence of verb menus, since the verbs are chosen contextually.

Sophia Hapgood in *Indiana Jones and the Fate of Atlantis* is a hybrid example. The game has three different modes in which it can be traversed: one is the most classic adventure game, which focuses on puzzle-solving; another is based on action, where there is fighting like in *Indiana Jones and the Last Crusade: The Graphic Adventure*, and includes some basic skill-based mini-games; the last one is the collaborative path, where the player can choose between Indiana Jones and Sophia Hapgood as player character depending on the part of the game. What is interesting is that Sophia is an NPC in the other two modes who accompanies Indiana but does not quite help, whereas in the collaborative path she is a sidekick assisting Indiana Jones while she is not being controlled by the player character.

Floyd, the robot in *Planetfall*, is one of the most notable NPCs in the history of adventure games (Murray, *Hamlet on the Holodeck*; Montfort, *Twisty Little Passages*). It is memorable precisely because of its nuanced reactions and behaviors to the player interaction. The following section reviews what made Floyd a great non-player character, and considers why there are not many characters like him in adventure games.

### **9.3.NPCs as Animated Characters**

Steve Meretzky, in his recounting of the creation of Floyd, lists four components involved in the writing of the character (“The Creation of Floyd the Robot”): direct interaction between player and character, reaction to player’s commands, independent character actions, and on-cue actions.

These components are the basic foundations of turning the NPC an animated entity. All these behaviors are simulated; what triggers them must be anticipated by the rules in the same way that behaviors had to be predicted in the case of game objects. A

lot of the personality of the NPC depends on their pre-scripted actions and reactions, just as in the case of the player character. Let us unpack what each of these design components means in terms of game design.

### **9.3.1. Direct Interaction between Player and Character**

NPCs are part of the world simulation—they are simulated characters. As such, the player should be able to interact with them as with the rest of the world, by using any of the verbs of the list. However, it is usually the case that only a few verbs will actually have an effect. As with the rest of the simulation, whether an action makes sense or not depends on the context. “Talk to” an NPC is a reasonable action, but “open” probably will not work. As we saw in the previous section, it turns out that a good deal of the interactions with the NPCs happen through dialogue. Thus the conversation system is the prime design device to establish the direct interaction between player and character. Lower levels of abstraction bring more possibilities to have a nuanced interaction with the character, since there is a wider range of verbs that can be used to interact with it.

In Meretzky’s example, if the player inputs “search Floyd,” the response will be the following:

“Floyd giggles and pushes you away. “You’re tickling Floyd!” He clutches at his side panels, laughing hysterically. Oil drops stream from his eyes” (“The Creation of Floyd the Robot” 137).

The output is a representation of Floyd’s behavior, which is quite human-like. What is remarkable is that an action like this, although it does not contribute to solving a puzzle, obtains a response that actually lets us know more about the NPC.

### 9.3.2. Reaction to Player's Commands

Since talking is the main interaction with the NPC, the relationship with it is mostly established through dialogue and writing, rather than actions. Replying to a question is the most basic type of reaction, although other reactions can relate to the mood and emotional state of the character. In *Planetfall*, the player can order Floyd to do something, which the robot may or may not do. Even in the case of Floyd not being able to perform the action, the response will again prompt a specific response to the command. Again using Merezky's example, if the player character commands Floyd to "walk north," this will be its response:

Floyd looks slightly embarrassed. "You know me and my sense of direction." Then he looks up at you with wide, trusting eyes. "Tell Floyd a story?" ("The Creation of Floyd the Robot" 137).

What is in effect an error message tells the player several things about Floyd: it has a bad sense of direction, it seems to like the player character, and it likes stories like a small child.

### 9.3.3. Independent Character Actions

Independent character actions are probably more common in text adventures, but not exclusive. These independent actions include moving around the world, and performing different actions without necessarily waiting for the player character to trigger them. The NPCs in *Deadline*, for example, have their own agendas, and will follow their daily routine as time passes, so that the player has to find where they are depending of the time of the day. Another example, rare in graphic adventure games, is Max in *Sam & Max Hit the Road*, who often seems to have an initiative of his own. While Sam looks for evidence, Max also looks around and plays with the objects in the

room: jumps on a bed, or tries to take an axe from the hands of a mannequin.

#### **9.3.4. On-cue Actions**

The NPC can perform pre-scripted actions in a given situation, pursuing its own agenda. These pre-scripted events give control to the designer, who dictates what will happen at a specific point of the game by having a computer-controlled agent to initiate the desired event. In the case of Floyd, the most significant on-cue action is its final self-sacrifice—Floyd offers to fight a large group of enemies to save the player character, and dies as a result (“The Creation of Floyd the Robot” 138). There is nothing the player can do to stop him, it is an on-cue action that is pre-scripted, because it brings the simulation to a specific game state. It is a way of introducing the designer’s intervention in order to facilitate the restoration of behavior without making the action seem arbitrary, since it is embodied in one of the characters in the game. The player has to generate the events of the story through her performance, while the NPCs can also perform their part in those events.

### **9.4.The Problems with Believable NPCs**

The believability of NPCs is relevant to providing engaging and responsive co-performers in the simulation. The historical trend in adventure games towards higher levels of abstraction has resulted in a reduction of the number of verbs. In that evolution, one of the first groups of verbs to be dropped from the verb list were meaningful actions that related to NPCs. Verbs that are available in textual interaction (e.g. kiss, hug, tickle, smile, punch, slap) do not appear in most verb menus, because they are not actions that can be used repeatedly in order to solve puzzles. However, they are actions that allowed

the player character establish a social relationship and potentially emotional ties with NPCs. Graphic adventures / point-and-click adventure games usually miss all the verbs that would help establish a relationship between player character and NPCs. To put it in contrast, these verbs are relatively important in MMOs, because they help establish social ties between players. In the long list of verbs in *World of Warcraft*<sup>47</sup>, we can find “clap,” “encourage,” “flirt,” “hold [somebody’s] hand,” or “roll eyes” in a long list of actions to display the player’s emotion and connect with players. The lack of this type of actions (*emotes* in MMOs) prevents creating an interactive social world in adventure games, particularly those with graphics.

NPCs in graphic adventure games have less self-initiative, so to speak, starting with their restricted movement. They often stay in one place, so that the player does not have to look for them; they basically behave like objects rather than characters. As Emily Short puts it: “An NPC who sits in a chair and does nothing is less convincingly alive than one who seems to be pursuing some sort of personal agenda.” As noted above, it is relatively easier to find characters who move around in text adventures, whereas they are relatively rare in games with graphics. Graphic adventure games usually leave their NPC animation in the hands the representation, rather than in their simulation.

Another reason for the absence of variation in the actions, reactions and movement of NPCs is that since those behaviors are represented, they must be accompanied of graphic and sound assets (e.g. facial expressions, animations, and voice reactions). Adventure games tend to have a large number of assets already; adding NPC behaviors makes the already crowded list of assets grow exponentially. It is easier to have

---

<sup>47</sup> See <http://www.worldofwarcraft.com/info/basics/emotes.html>

NPCs with more complex behaviors in text adventures, since the implementation is relatively easier—the asset may be a single line of code that will print the description of the behavior. In fact, the most relevant tutorials and experiments with NPC design all refer to Interactive Fiction (Atkinson; Meretzky “The Creation of Floyd the Robot”; Roberts; Short; Wood).

The excuse for this sacrifice may be the high difficulty to design a believable, nuanced character. Even in the case of Floyd, there will always be reactions that seem out of place, and not human-like. As Steve Meretzky again puts it:

“The only area where I would have liked to see continued improvement was in the area of talking to NPCs. But the main problem with making NPCs seem more deep and real wasn’t due to parser limitations, it was just the sheer amount of work needed to give a character enough different responses to keep that character from seeming “canned,” even for a short while” (Rouse 183).

The variation in the responses to player interaction and dialogue is relatively easy to fake, by writing different ways in which a reply will be expressed. These responses can be displayed randomly, or conditionally depending on the context (e.g. if the same topic has been asked about before). However, as long as the responses are written (and/or spoken), even after writing a wide range of them, the artifice will eventually be exposed, and the responses will repeat. In the same way that game designers need to anticipate all the possible actions the player will try to perform with objects, anticipating all the possible situations that the NPC will have to respond to can be a Sisyphean effort. Simulating human likeness is even more complex than creating a computer version of a phone box, because is far easier for the player to “break” the character and reveal its artificiality.

Two of the disciplines that can tackle the problem of NPC responses are artificial

intelligence and computational linguistics. The problem with NPCs is less an exclusive issue of adventure games themselves, and more a problem of those disciplines. Game design has come up with a few ways to “fake” human-like behavior based on current technologies; further advances in computer science will help game design find new and better solutions.

There are aspects that seem to be within the scope of game design, however, that adventure games could tackle. For instance, there are not many examples where mood shapes the behavior of the characters. In the context of IF, Short mentions “mood and emotion” as one of the ways to give personality to an NPC, but does not mention any examples. Another aspect that could be explored is that of social relationships between characters, where whether a character likes another character or not affects its behavior. For example, it may be more likely for the player character to obtain information from a character that finds her sympathetic; or if the player has upset an NPC, the reaction of its sibling may be negative. These experiments are within the possibilities of design, and although there will be limitations, we will not know what those limitations are without further experimentation.

Having the believability of NPCs rely on representation exclusively is somewhat of a cop-out, and does not quite take advantage of the procedural properties of the medium upon which the simulation is built. It is true that it is very difficult to create believable NPCs, but current commercial adventure games seem to be too complacent about not dealing with the problem at all, resorting to previous design solutions in their new games. There is still hope, however, since the authors of non-commercial games (particularly Interactive Fiction) seem willing to tackle the problem, even if their



resources are limited.<sup>48</sup>

### **9.5.Conclusion**

Since non-player characters are simulated, they encounter some of the same shortcomings that affect other parts of the simulation. However, the problem is more complex than making a treasure chest that opens, closes, and contains golden coins—NPCs must be give the appearance of life. Although adventure games (especially text adventures) have tackled it from several fronts, it seems that the most abstracted, less simulatory solutions have prevailed. The way to circumvent these problems has been to resort to traditional dramatic writing, which already has a catalog of devices to bring characters to life. However, there is a good range of issues that adventure games could still tackle—NPCs with more self-initiative, who navigate the world and act based on pre-scripted patterns; innovative dialogue systems; introducing mood as a design factor in a character. More complex NPCs would give more depth and complexity to the simulated world, and would also facilitate the restoration of behavior, since the player could count with worthy, believable co-performers.

---

<sup>48</sup> See Short, Roberts, and discussions about NPCs in *rec.arts.int-fiction*.

## **CHAPTER 10**

### **TEMPORAL FRAMEWORK OF ADVENTURE GAMES**

#### **10.1.Regulating Time**

Performances, narratives, and gameplay all take place over time. In all of them, there is a recurring dichotomy between the internal time of each medium (the time regulated by the performance or the game, the time as represented in the narrative), and the time outside of it, in the “real world.” The relationship between internal and external time is not necessarily conflictive, but the extent to which they coincide or diverge creates a series of dynamics that give way to different ritual forms, narrative devices, or, more relevant to this dissertation, game design devices.

The concept of time that we are dealing with here is experiential: how time passes for the player during gameplay according to the rules of the game. In performance activities, the rules establish the relationship between the internal and external time. Thus rules set the pace of the performance (of the game in our case), whereas in traditional narratives (e.g. prose fiction or film) the pacing is reflected in the representation .

This chapter deals with the different spheres involved in regulating time in games (real time, game and story), and proposes a time framework for adventure games. We argue that story is a foundational component of the temporal framework, rather than sidestepping it or considering it an exception. The first step in setting up the framework is to survey how time is regulated and experienced in performance, narratives, and videogames.

## 10.2. Performance Time

Time is one of the basic qualities of a performance, as we saw in Chapter 4. Performance takes place over time, as Schechner points out, and performance time is regulated by the activity itself (*Performance Theory* 8). As we described in Chapter 4, there are three varieties of performance time, according to Schechner (8-10): event time, set time, and symbolic time. Performance activities, including videogames, can combine more than one type of time regulation. Let us revisit these terms again.

### 10.2.1. Event Time

The length of the performance is determined by however long it takes to complete a series of set steps. Examples of performances where time is determined by a series of events are weddings, sports such as tennis or races, most theatre plays, and games such as hopscotch. Event time usually implies that there is a specific order in which those steps must be taken, but there is no time constraint in which to do so. Event time is also fairly common in videogames, where the player has to fulfill pre-established goals in order to complete or win the game. Videogame racing games are one example; action role-playing games such as *The Legend of Zelda* series require the player to succeed in a series of quests.

The order in which the steps are taken seems to be a factor in determining how “linear” a game is. At times, there may be only one order in which those steps can be taken, because it is part of the story, or because there is a certain progression that is designed into the game. This is the case of *The Legend of Zelda: Phantom Hourglass*, where the player has to complete the dungeons in a certain order; the player character Link obtains an object whenever a boss is defeated, and that object allows the player to

reach and be able to traverse the next dungeon.

We can understand the sets of steps as a series of goals that can be organized sequentially, requiring the player to play the game in a linear fashion. Goals are usually interdependent in order to mark a progression in the game; progression encourages the player to keep playing, because it reflects how she is “getting better” at the game.

Another way connecting the steps taking place over time is to establish a causal connection between them—completing one set of steps causes something else to take place, which becomes the next situation the player has to deal with. As we saw, this is a more common strategy in adventure games—since they are puzzle-based, it may be more difficult to give the player the sense of improving in the game. Thus adventure games use causality to mark progress, exploiting their story to provide the continuity between challenges. The presence of this continuity is what differentiates adventure games from puzzle games with a story setting, such as 7<sup>th</sup> *Guest* or *Professor Layton and the Curious Village*.

The goals may be multisequential, i.e. the player may be able to complete the goals in any order, so that the pre-supposed unisequential nature is subverted. There can still be progression, since there are a specific number of goals to be attained; as we saw in Chapter 6 these goals are set up in clusters, which the player can complete in any order. For example, in *Myst*, the player does not have to follow a specific order to solve the puzzles in each Age. Although there is not a causal relationship between puzzles (the puzzle-solving events in *Myst* make up a weak story after all), the puzzles are still concatenated, since the outcome of solving one puzzle usually helps solve the next.

### 10.2.2. Set Time

When there is a set time for a performance, it means that there is a specific amount of time in which the activity can take place, whether it is completed or not. Many sports use time periods as a way to regulate their time, such as basketball or football. Whoever scores the most within the allotted time wins the game. Videogames can use this model in competitive, multiplayer modes, e.g. the “shine thief” mode in *Mario Kart: Double Dash*, where four players compete to keep an item called a “shine” (a token in the shape of a sun). Whoever keeps the shine at the end of sixty seconds is the winner of the game. As we will see below, adventure games do not use set time in their general design, but only for specific puzzles or in mini-games.

### 10.2.3. Symbolic Time

Performance time can represent some other period of time, either longer or shorter. For example, the events of Shakespeare’s *A Midsummer Night’s Dream* take place during a whole day, but the performance may only take 2-3 hours. When the time represented within the performance and the time outside of it are equivalent (1:1), we can say that the performance takes place in “real time.” Games can use symbolic time as a device; for example in *The Legend of Zelda: Majora’s Mask*, the player relives the events that take place within a three-day cycle. Three days in-game, however, are roughly one hour of the player’s time.

The concept of symbolic time is key to understanding the relationship between time in narratives and videogames. The discrepancy between the time represented within the performance creates a gap that both narratives and videogames can take advantage of in order to pace the reader / player experience. Time flows steadily in the real world; the

representation of time in any of these media influences how the time outside of them is experienced. As we saw above, while in traditional narratives the representation establishes the pacing, in the performance the pace is set by the rules.

### 10.3.Narrative Time

Time is also a factor in the story of the game, so we will therefore deal with it separately from performance. Represented time is key to understanding the basic duality of narrative time. Genette quotes Christian Metz and Gunter Müller when he establishes the dichotomy between story time (“the time of the thing told” according to Metz, the time represented in the narrative) and narrative time (the actual time it takes to read, to traverse the written narrative (Genette 33-34). As we saw above, this dichotomy is easier to grasp in performance activities, which are by nature time-bound, although it also occurs in literature. For example, Isaac Asimov’s novel *Foundation* covers the last 175 years of the Galactic Empire in less than 300 pages; obviously, it does not take 175 years to read, although every reader will need a different length time to traverse it. Represented time can also be expanded: think of all the movies where a bomb is about to explode within 30 seconds, which may end up being more like five minutes of film length.

In traditional narratives, there may be a digression between the order of the events in the story as they happen, and the order in which they are told. Flashbacks (*analepsis*, in Genette’s terms) and flash-forwards (*prolepsis*) are a common narrative device. Games use these narrative devices in cut-scenes, rather than making the change of narrative order into a participatory event. Juul argues that the order of events in a game is problematic, since their causal relationship may be broken—the actions of the player would not make sense in an interactive flashback (*Half-Real* 147-8).

However, rather than discarding it, interactive flashbacks and flash-forwards are an interesting design challenge for story-driven games. As we saw, the gameplay of adventure games is already segmented, so the order in which certain events happen and have to be restored can be altered. Interactive Fiction has already tackled this—*Tapestry* has the player control a man who just died, who has to go back in time to fix the mistakes of his past. In this case, the past becomes separate diegeses, which the player revisits in turns. By solving them, the player is supposed to affect the destiny of the player character. Segmenting the game into chapters can also introduce interactive flashbacks or flash-forwards—a couple of chapters in *Indigo Prophecy* / *Fahrenheit* are flashbacks in the story of the protagonist that the player has to re-enact. The issues Juul seems to have with flashbacks and flash-forwards come from the belief that the events can only be generated by the interaction with the rules of the game, discounting that restoring a specific behavior, which would not alter the relations of causality between events, can also be a gameplay activity.

#### **10.4. Time in Videogames**

The dichotomy between external and internal time also extends to videogames. In this section, we will briefly lay out a basic framework of how time is regulated in videogames, based on Juul's description of game time in *Half Real* and Zagal and Mateas' "Temporal Frames: A Unifying Framework for the Analysis of Game Temporality." Whereas the first lays the foundation of a basic theory of time in games, the second proposes understanding time in games based on different frames, different interrelated spheres where time passes and is managed differently. Juul describes this as the duality that involves play time and fictional time, whereas Zagal and Mateas

introduce two similar concepts, real-world time and gameworld time.

#### **10.4.1. Play Time / Real World Time**

Play time, according to Juul, is the time the player takes to play; it is the time of the player while she is interacting with the game. Zagal and Mateas understand this type of time as “real-world” time, which refers to the frame of the physical world of the player. The distinction between this concept and play time is that play time does not explicitly include events that may be related to real-world time, such as a specific time of the day or the year. This specific extension of the concept allows Zagal and Mateas to account for phenomena such as the events in *Animal Crossing* that are triggered by real-world time, such as Christmas celebrations.

#### **10.4.2. Fictional Time / Game World Time**

Fictional time is the time taken in simulation, which is different from the time spent interacting with the game at an extradiegetic level, e.g. saving your game, changing the game settings, or watching a recording of your interaction with the game (e.g. watching the replay of the race you just ran in *Mario Kart: Double Dash*). Juul does not have a term to refer to the time where the player is interacting but not playing, although he refers to it in his breakdown of the possible asynchrony between play time and event time; Zagal and Mateas do not account for it at all.

Zagal and Mateas refer to this temporal frame as *gameworld*<sup>49</sup> time, which is “established by the set of events taking place within the represented gameworld.” In this

---

<sup>49</sup> Zagal and Mateas’ use of the term *gameworld* is roughly equivalent to how we have used *simulation* throughout this dissertation.



sense, gameworld time can have its own way to establish cycles and duration of activities, such as the day/night cycles, and have those cycles and durations be relevant to the game design. The labels applied to those cycles are “fictive time,” which is the only term in their framework addressing the time of the narrative (this will be discussed below).

With gameworld time, Zagal and Mateas seem to obviate the question of whether a game can be understood in terms of having a gameworld or not. According to the methodology of the Game Ontology (Zagal et al.), which is the foundation of their temporal framework, there are stronger and weaker examples of gameworld, so that there is always some sort of implied gameworld even in abstract games such as *Tetris*. Conversely Juul notes that in abstract games there is no gameworld to speak of, because they do not include a sense of play-pretense. Thus in abstract games, the player is interacting with the game as a state machine, so that time in the game passes as the game state changes after each interaction (*Half-Real* 142). Juul implies that fictional time usually happens in games that can be understood in terms of gameworld, since having a gameworld is also a requisite for games that have a story component. Thus the relationship between how time passes in the gameworld seems to be strongly connected to the fiction of the game.

The concepts of Play Time and Fictional Time allow us to establish the parallels between performance, narratives, and videogames, and how the internal time of the activity relates to the time outside of it (see Figure 17).

	External	Internal
Performance	Real Time	Symbolic Time
Narrative	Narrative Time	Story Time
Videogames	Play Time	Fictional Time

*Figure 17: Dichotomy between time inside the media form and outside of it.*

### 10.4.3. Fictive Time

Zagal and Mateas refer to another temporal frame, Fictive Time, to refer to the “socio-cultural labels to a subset of events.”<sup>50</sup> This frame seems to include non-interactive sections of the game (e.g. cut-scenes, dialogue, flashbacks); it is somewhat weakly defined. The term seems to lump together both how time periods are named or referred to and the “narrative” of the game, avoiding any problematic references to how a narrative may be integrated within gameplay.

Juul does not distinguish the time of the gameworld from the time of the story, although he accounts for it as he explains the different ways in which play time and fictional time map onto each other. Juul explains how a cut-scene interrupts play time (since the player is not interacting with the game while they are playing), but actually is part of fictional time, since the events shown in the cut-scene are happening in the

---

<sup>50</sup> Zagal and Mateas also include Coordination Time, a frame that refers to how the time to play is organized, e.g. in turns, rounds or set sequences. Since this category seems to refer to multiplayer games, and adventure games are typically single player, we will not take this frame into consideration.

gameworld.

Both Juul and Zagal evade talking about how the story may be integrated in the game, or how its temporality and order of events may relate to the time of gameplay. We have already discussed how the relationship between stories and games is problematic, and is not the focus of these authors. On the other hand, it is an issue that we cannot avoid in the study of the adventure game genre, since gameplay and story are so strongly connected in it.

### 10.5.Temporal Framework for Adventure Games

In order to study how time figures in the story and the game as performance, we need a framework that includes a time frame for both. This framework is based on the terms proposed by Juul and Zagal, and applied to adventure games. The framework has three distinct components: player time, simulation time, and story time.

In adventure games, when time passes in the three spheres simultaneously, the player is effectively restoring the behavior in the game by solving the puzzles, progressing by fulfilling the goals set by the designer(s) of the game (see Figure 18).



*Figure 18: Temporal Framework of Adventure Games.*

### 10.5.1. Player Time

Player time refers to the time that the player spends interacting with the game, as Juul defined it. This interaction includes performing in the diegesis (playing the game), as well as extradiegetic performance (calling up menu options such as game settings and saving / loading the game). When the game options are called up, time usually stops in the other two frames (simulation and story). This time is extradiegetic, i.e. outside of both the simulation and the story. Time passes chronologically, but it only counts as play time as long as the player is interacting with the game.

### 10.5.2. Simulation Time

Simulation time in adventure games is best understood by combining Zagal's gameworld time with Juul's fictional time. This is the time of the events that happen within the simulation, as the player interacts with it. These events are diegetic; they happen in the simulated world, but not all of them may be part of the story. The player can interact with the simulation without making time pass in the story, because the player may not be solving the puzzles—this is the time for exploration.

Allowing the player to explore the world, without advancing the story, is one of the things that adventure games can do best. Adventure games can present a rich simulated world, where the player can do things that are neither part of a puzzle or advance the story, and yet let the player learn more about the fictional world and its pre-history. Even if it is not relevant to puzzle-solving, exploration contributes to the nuance and complexity of the simulation. For example, in *Monkey Island 2: Lechuck's Revenge*, a set of puzzles requires the player to check out three books from the library. The player can check the whole library catalog, which is full of joke titles, until she finds the right

books, or get the titles from talking to other characters. The player may browse the catalog at leisure even after solving the puzzle; it is time spent interacting with the simulation, but they are not events of the embedded story.

How time passes in the simulation is regulated by the rules, which establish what the different cycles are. This regulation will be dealt with in the section regarding passage of time as a design element below.

### **10.5.3. Story Time**

Time also passes in the story of the game, which is different from the other two frames. This frame refers specifically to the embedded story, the series of goals that must be fulfilled in order to complete the game. It does not include the pre-history, but to what the player must do in the game in order to restore the behavior expected in the game. As Juul notes, time passes in the fictional world during cut-scenes, or the equivalent representation of events—the embedded story unfolds, even if the player does not have control over the events (*Half-Real*, 145-7).

This division between what the player is expected to do and what the player does is relatively clear in adventure games, but it is not the case in other genres where the play-history is not so constrained by the mechanics of the game. For example, in role-playing games, the narrative may require the player to obtain an object by purchasing it. How the money is obtained, however, is up to the player and how she interacts within the rules of the game: the player can buy and sell objects, complete another quest and get paid for it, steal it from other characters, or by looting defeated enemies.

The story frame is key to understanding how time passes and is regulated in adventure games. The intersection between player time, simulation time, and story time

means that the player is doing what the designer(s) of the game intended, but there is no room to advance time in the story other than puzzle-solving. What is more, since the puzzles can only be solved in the way(s) specified by the rules, there is usually only one way to advance the story. The player can (and has to) spend time exploring the world, but time will not pass in the story. This is one of the features that is usually decried by those who advocate new storytelling forms in games (see Spector): adventure games seem to have no room for emergent storytelling, which in relation to time should be the intersection between play time, simulation time and story time. In adventure games, that intersection is occupied by puzzle-solving.

Adventure games also present interesting temporal phenomena in the intersection between story and play time. How does time pass in the story and play time frames, but not in the simulation? My proposal is that this intersection is the domain of experimentation in the game, of trying things to see what happens without saving the game—this is the domain of ontological manipulation. Rau refers to this time as “reload time,” when the player learns something about the game after making a mistake, and may load a save state to apply that new knowledge. Ontological manipulation is allowed by the “undo” command in some Interactive Fiction; it is one of the main game mechanics of *The Last Express*, where the player can rewind time as much as she likes, and do things differently. This is “what if...” time, where the story differs for a bit from the expected, ideal walkthrough. This series of moments takes place in an alternate timeline of the story of the game, albeit a brief one, since they are decisions that will not allow the player to finish the game.

## 10.6. Passage of Time as a Design Element

As mentioned above, the regulation of time in the simulation is part of the design of adventure games. How game design regulates time in the simulation depends on the reference frame used to determine how time passes: the player (real time), the simulation (cycle-driven time), or the story (event time).

The passage of time is an added design element to the puzzle-based nature of adventure games. Time constraints are not an inherent part of puzzle-solving—puzzles usually do not require the player to solve them within a time limit, thus allowing for a more sedate pacing during gameplay. However, since time prefigures importantly both in games and stories, puzzles in adventure games can make use of time as a way to increase challenge or create new ones.

### 10.6.1. Real Time

Time in the game passes in real time when simulation time is synchronized with the player's time, i.e. time passes no matter what. Real time as a design device is normally used for specific puzzles or sections of a game. When it is used, it usually implies there is a time limit within which the player has to act. For example, in *Space Quest: The Sarien Encounter*, Roger Wilco is chased by a spider droid in the desert, which will do its best to hunt him down. The player must have Roger move to avoid the droid in real time; in order to get rid of the droid, the player must wait for it to move under a bridge, and then make the bridge collapse at the right moment.

Such real-time activities usually require some basic skill and reflexes, rather than puzzle-solving. In some games, the sections that use real-time are closer to mini-games, where the usual gameplay is also interrupted. This is the case of the mini-games in

*Indiana Jones and the Fate of Atlantis*, which includes a car chase and a submarine piloting mini-game.

Introducing real time can affect the puzzle in unexpected ways. Anja Rau describes this design choice with another example from *Space Quest: The Sarien Encounter*. One of the puzzles requires Roger Wilco to walk through a tunnel whose ceiling is dripping acid, so the player has to move Roger avoiding the drops. The acid falls at a fixed rate; however, the game offers the option to speed up the character animation, which makes it easier to get across the tunnel. The acid falls in real time, but the animation runs according to the game settings (Rau, “Reload: Yes/No”).

In *The Last Express*, most of the events in the game take place in real time, accelerated by a factor of six (Remo). It is considered real time because there is no stopping it, events take place following a specific pre-established script, whether the player is there to witness them or not—the time of the player, the simulation, and the story is passing simultaneously. This generates a specific pace in the game—the first few hours feel slow, with periods of “dead time” where nothing happens, whereas the accelerated passage of time precipitates events towards the end. This pacing of events also invites the player to explore the space continuously, which keeps changing as the game advances. Scheduled events can give clues about where the player should go—for example, the dining car is busy during meal times, so it is a good occasion to meet people, or to go to their compartments to search them while they are away. The train schedule also marks transitory goals; for instance, when the train stops at Epernay, the player character has to remain hidden while the police are searching for him. *The Last Express* is an experiment with game time in all three spheres, and an example of the



design possibilities of synchronizing them.

### 10.6.2. Cycle-driven Time

In the early days of text adventures, time in the simulation was regulated by cycles, so this type of time regulation is exclusive of text adventure games. We introduced Montfort's definition a cycle in Chapter 2: one input from the player and its corresponding output, until the next input from the player (Montfort, *Twisty Little Passages* 25). Cycle-driven time takes the simulation frame as its main reference, so that time passes after each player interaction. Juul refers to how in abstract games the passage of time was marked by the interactions with the game as a state machine (*Half-Real* 142); cycle time is marked precisely by each player interaction.<sup>51</sup>

In the original *Zork*, time is regulated by a clock demon, which is a mechanism that schedules "arbitrary events for arbitrary future times" (Lebling, Blank, and Anderson 55, 58). Such events can be how long the lamp can burn, or when wounds heal. Significantly, the clock does not advance if there is unparsable input, i.e. input that the program cannot process because the terms are not part of the vocabulary, or because the player made a typo. Thus time would only pass if the player's input is valid, even if it "nothing happens" (55).

In many text adventure games, every cycle would represent a specific period of time. This representation could be explicit, so that each cycle has a "label," as Zagal and Mateas mentioned. For instance, in Infocom's *Deadline*, each cycle represents a minute in the simulation; if the player chooses to "wait" for a cycle, it will be equivalent to three

---

<sup>51</sup> As we saw, graphic adventures have inherited the cycle-driven interaction, although they do not use cycles to regulate time, using real time as a design element instead.

minutes. This time representation is significant to gameplay—the player has to solve a murder mystery in twelve hours of simulation time.

*The Hobbit* presents a variation on cycles—although time is still cycle-based, if the player does not input a command for some time, the game will assume it means “wait,” having real time make game time pass. Early on, the player character has to escape the attack of a group of trolls—if the player spends too much time thinking what to do, time will pass and the trolls will kill Bilbo, the player character. Hiding and letting time pass is the way to defeat the trolls, since with the arrival of dawn they will turn into stone.

Some games only used cycle time in certain sections or puzzles. This is true, for instance, whenever the player character engages in combat: in *Zork* the player only has a couple cycles to kill the troll. Cycles can also establish a time limit increasing the difficulty of a specific puzzle. The notorious Babelfish puzzle in *Hitchhiker’s Guide to the Galaxy* is such an example—the solution requires performing a very specific set of actions in a very specific number of turns, with no mistakes.

### **10.6.3. Event-driven Time**

The passage of time can also be driven by the events in the story, either by solving puzzles or in the cut-scenes. This is by far the most common regulation of time in adventure games, since nothing happens until the player solves a puzzle. Because there is no implicit time limit, the player can take her time to think about how to solve a puzzle or to explore the world, since time is not passing until she solves puzzles.

Time limits were far more common in text adventures and earlier graphic adventures; now they are seldom found outside of mini-games, perhaps due to the

influence of *Myst*, or to the tendency to think of adventure games as the opposite of action games. Thus adventure games tend to have a more sedate pace than most other genres, particularly when the player character cannot die, since there is no pressure and no risks involved in trying things. Event-driven time means that the player can set the pace of her own gameplay.

*The Secret of Monkey Island* exemplifies how event-driven time works. In the first part of the game, the player character has to pass three trials to become a pirate. After completing those, Governor Elaine Marley is kidnapped, so the next goal in the game is to rescue her. The island of Melee also changes—the pirates have deserted the bar, and are out pillaging the Governor’s house now that she has gone. The kidnapping does not have a direct causal relationship with completing the trials, but it is the event triggered after the first set of goals in the game are completed.

### **10.7.Conclusion**

The regulation of time in adventure games has transformed with the improvement of technologies, and the change of interfaces, from textual to graphical. Although the interaction in adventure games is still based on cycles, the regulation of time can shift depending on which sphere is regulating it (the player, the simulation, or the story). The regulation of time can shift even within the same game depending on each puzzle. Mini-games also can bring about their own time regulation, since they have rules that may be different from the rest of the game.

Designing how time passes in an adventure game is one of the cornerstones of the gameplay experience. It is also a design element that is increasingly overlooked, since later adventure games tend to eschew time limits in order to provide a slower-paced type

of game, with less pressure for the player. This sets adventure games apart from other genres—players can set their own pace, rather than having to keep up with the game.

## CHAPTER 11

### CONCLUSION

#### 11.1. Reconciling Story and Game through Performance

The hybrid nature of adventure games, which have been referred to as “story-games” (Buckles, “The Computer Storygame *Adventure*”; Aarseth, “Genre Trouble”), allows us to understand better how games and stories can be reconciled. Adventure games implement a fictional world, host to the story, in the form of a simulation. Through her participation in that simulation, the player becomes a performer. In adventure games, the player must re-enact the events of the embedded story; thus the main goal of the game design is providing the player with the means to carry out that re-enactment in order to traverse the game successfully.

The formal elements of adventure games are therefore at the service of the player, so she can traverse the game successfully. The player character determines how the player can interact with the simulation, becoming the locus of the game mechanics. The range of actions the character can perform establishes how the player can manipulate the objects in the simulation. The player character provides the means to solve puzzles, which is the main gameplay activity of adventure games. Puzzle-solving is key to restoring behavior in videogames, because it is what changes the game state; the puzzles are also part of the fictional world, so every solution also brings about an event or series of events in the story of the game. Since the solutions to every puzzle are usually anticipated in the game design, the actions of the player usually have to follow a specific set of steps; those steps are the script that must be restored through the performance of

the player.

The constraints the game imposes on the player create the perception that an adventure game is “linear.” However, the player often has leeway to interact with the game. As we saw, some games include puzzles that do not need to be solved to traverse the game successfully. More importantly, the player must explore the simulation, in order to figure out its workings. Exploration also provides the player with information about the fictional world, which contains the domain knowledge necessary to solve the puzzles in the world.

A key affordance of the simulation of adventure games is encouraging exploration and experimentation. A more elaborate simulation provides the world with depth, because it uses lower levels of abstraction in the implementation of the fictional world—there is more to do in the simulation, and there are more ways to do it. On the one hand, lower levels of abstraction encourage the player to discover the fictional world. On the other, a deeper world may also obscure the goals of the game, since there are more possible actions that do not restore the implicit behavior of the game. Thus adventure games have tended towards higher levels of abstraction in the model of the fictional world, since lower fidelity in the simulation curbs what the player can do, and that makes it easier for the player to restore the proper behavior.

The evolution of adventure games has therefore witnessed consecutive trade-offs, moving towards higher levels of abstraction so that the player does not stray too far from the behavior to be restored. This trend has also meant less complex simulations, weakening one of the features that set adventure games apart by representing the fictional world rather than simulating it. In the end, these design decisions have privileged the

story over the depth of the simulation.

## **11.2. Lessons Learned from Adventure Games**

There are many things to be learned from adventure games, both from their limitations and their achievements. These lessons, which can be applied to other videogame genres, can be taken from several design aspects: creation of fictional worlds, integration of the story into the space, scripting the interactor through the design of the player character, and NPC design.

### **11.2.1. Creation of Fictional Worlds**

Adventure games may have been introduced as “interactive fiction,” but what they do best is creating worlds. As we saw, the simulation creates the diegesis that the player interacts with. Earlier adventure games tackled complex simulations where the player could explore and try things. The problem with those simulations is that it takes a long time to implement them, and to anticipate all the possible actions the player may take. Advancements in technology may help creating more nuanced simulations, but it is still a game design problem.

Rich simulations encourage exploration and experimentation, inviting the player to inhabit the simulated world. Learning the story of the world, finding out all its secrets, is part of both the puzzle-solving (as a source of information) and the pleasures of the game—exploration is an underrated quality in videogames in general.

A large simulation can become a problem if the goal of the game is completing the steps that make up the events of a story. Adventure games have provided recurring examples of how to “script the interactor,” in order to guide the player to the goal, but

without restricting the actions in the world.

### **11.2.2. Scripting the Interactor through the Design of the Player Character**

The player character is another design element that designers of other videogame genres can learn from, particularly in games where the personality of the player character is defined and distinct from the player. The player character is the main device to script the interactor (Murray 79), since it is a game entity that defines which actions the player can perform in the game. Action also defines the character, and even though there is no role-playing involved, the verb list makes it easier for the player to behave like the character. In some cases, as we saw with *Monkey Island 2*, the player has to think like the character in order to solve puzzles.

A range of strategies to script the interactor can help the player restore the behavior implied by the game, without coercing her to perform in a specific way. These strategies go from associating the character with a specific type of genre fiction (e.g. detective novels, folktales), to selecting the type of actions that will generate the proper behavior (e.g. cross-questioning vs. fighting). Scripting the interactor efficiently helps create the illusion of control on the part of the player, since it encourages the player to discover what the proper script of the game is.

### **11.2.3. Integration of the Story and Puzzles in the Space**

A feature that originated in adventure games is the inscription of the story in the space. The descriptions of objects in text adventure games have endured to this day, since they can provide information that graphics alone may not be able to convey. The physical space is the first thing that is simulated in the game, the objects and the characters



contained in it are both the existents of the story and the objects and entities in the game. The space is not only where the events take place, it also has its own story which the player can discover as she plays. Inscribing the story in the space through descriptions, documents, indexical storytelling, and micro-narratives are all strategies that can also be used in other genres, since the construction of the space is a foundation of game design.

The puzzles involve the existents of the story, thus the solution to a puzzle and the steps to achieve it are also events in the story. Therefore, as the player solves the puzzles, she is also advancing the story. This is achieved through the integration of the puzzles in the fictional world. In adventure games, the story is not a reward for overcoming a challenge—it unfolds as the player restores the behavior, as the actions involved in puzzle-solving or as represented events.

#### **11.2.4. NPC Design**

NPCs in adventure games, as in most other genres, are not quite human-like. Commercial adventure games in the last ten years have opted to implement NPCs that function like signposts, giving specific information but not allowing the player to interact with them, lest the limitations of the design reveal that the character is a program.

However, it is also true that text adventure games have provided a few examples of accomplished NPCs, whom the player can not only talk to or fight, but also kiss, tickle, smile at, or push around, and get an appropriate response. Some text adventure games also provide autonomous characters, who move in the world pursuing their own agendas. The challenge of designing believable NPCs has been dropped too soon, and it does not seem that other genres are picking up the slack to create dramatically sound characters. Dramatic writing has substituted a more nuanced simulation of character, so that

interacting with an NPC usually means triggering a dialogue cut-scene.

Adventure game NPCs are a specific example of how improvements in the technology do not automatically bring about more complex simulations. Even though computers now have the processing power, the graphics, and the storage needed to create characters who react to the player's input and also have self-initiative, adventure game NPCs are simulated using high levels of abstraction, or are just represented. Better technology also means it is more complex to create NPCs, since there are more assets to produce (e.g. 3D models, animations, voice recordings). Representing the character, rather than simulating it, also means that there are fewer ways to break down the illusion of liveliness; characters are particularly susceptible to reveal their artificiality in the interaction. Since better technology has given rise to more verisimilitude in the representation, the simulation has to use higher levels of abstraction in order to maintain the illusion.

### **11.3. Contribution to Game Studies**

The goal of this dissertation has been to help situate adventure games within game studies. Adventure games are a legitimate subject of study in this academic field, since they implement their fictional world in the form of a simulation.

There are several factors that may prevent identifying the simulation of adventure games. The predominance of represented content can obfuscate the properties of the simulation. That is, the aspects of the fictional world that are not simulated, such as cut-scenes or lengthy descriptions, can generate the perception that adventure games are not simulations, as Aarseth misleadingly affirms ("Genre Trouble," "Doors and Perception").

Another factor that can undermine the simulation of adventure games is the trend

towards higher levels of abstraction, which both reduces the functionality of the simulation and limits what the player can do in it. As we have seen, this is a design decision that is meant to help the player restore behavior.

The most important contribution of this dissertation is the application of the concept of restoration of behavior to videogames, to account for the player as a performer and as the activity that brings together story and game. Every videogame encourages a different type of behavior, determined by its game mechanics. In the case of adventure games, the restoration of behavior must take place in a specific way, so that the player enacts the events of the embedded story.

Adventure games have provided a productive case study to examine the problematic relationship between games and stories. Reconciling both depends on how the player performs in the simulation, since it is the player who re-enacts the story events as she interacts with the game. Adventure games have cultivated the art of facilitating this re-enactment for more than thirty years. This dissertation reveals how much there is to learn from this rich history, which can inform both our understanding of the genre and of videogames in general.

## REFERENCES

- “Shakespeare’s Globe: Theatre”. 2009. <<http://www.shakespeares-globe.org/press/theatre/>> (accessed November 8, 2009).
- Aarseth, Espen J. *Cybertext*. Baltimore, MD: Johns Hopkins University Press, 1997.
- . “Computer Game Studies, Year One.” *Game Studies* 1.1 (2001). September 2009  
<<http://www.gamestudies.org/0101/editorial.html>> (accessed November 8, 2009).
- . “Genre Trouble: Narrativism and the Art of Simulation.” *First Person : New Media as Story, Performance, and Game*. Eds. Noah Wardrip-Fruin and Pat Harrigan. Cambridge, MA.: MIT Press, 2004. xiii, 331 p.
- . "Doors and Perception: Fiction Vs. Simulation in Games." *Digital Arts and Culture*.
- Abbott, Edwin A. *Flatland - A Romance of Many Dimensions*. (1884). Signet Classic New York: Signet Classic, 1984.
- Adams, Ernest, and Andrew Rollings. *Fundamentals of Game Design*. Prentice Hall, 2006.
- Aristotle. *Poetics*. Trans. Malcolm Heath. Penguin Classics. London: Penguin Books, 1996.
- Artaud, Antonin. *The Theatre and Its Double*. Trans. Victor Corti: Calder, 1970.
- Asimov, Isaac. *Foundation*. Gnome Press, 1951.
- Atkinson, Doug. “Character Gender in Interactive Fiction.” *XYZZY News*.3.  
<<http://www.xyzzynews.com/xyzzzy.3h.html>> (accessed November 8, 2009).
- . “Character Gender in Interactive Fiction Part II.” *XYZZY News*.  
<<http://www.xyzzynews.com/xyzzzy.6f.html>> (accessed November 8, 2009).

- Baldick, Chris. *Concise Dictionary of Literary Terms*. Oxford: Oxford University Press, 2001.
- Bartle, Richard. "Hearts, Clubs, Diamonds, Spades: Players Who Suit Muds." *The Game Design Reader: A Rules of Play Anthology*. Eds. Katie Salen and Eric Zimmerman. Cambridge, MA.: MIT Press, 2006.
- . "Making Places." *Space Time Play - Computer Games, Architecture and Urbanism: The Next Level*. Eds. Friedrich von Borries, Steffen P. Walz and Matthias Bottger. Basel: Birkhäuser Verlag, 2007.
- Barton, Matt. *Dungeons and Desktops: The History of Computer Role-Playing Games*. Wellesley, MA: A K Peters, 2008.
- Bates, Bob. *Game Design*. 2nd ed: Thompson Course Technology, 2004.
- Baudrillard, Jean. *Simulacra and Simulation*. (1981). Trans. Sheila Faria Glaser: The University of Michigan, 1994.
- Boal, Augusto. "Theatre of the Oppressed." *The New Media Reader*. Eds. Noah Wardrip-Fruin and Nick Montfort. Cambridge, MA.: MIT Press, 2003.
- Bogost, Ian. "Performative Play." *Gamasutra* (2008).  
<[http://www.gamasutra.com/view/feature/3703/persuasive\\_games\\_performative\\_play.php](http://www.gamasutra.com/view/feature/3703/persuasive_games_performative_play.php)> (accessed November 8, 2009).
- Borries, Friedrich von, Steffen P. Walz, and Matthias Bottger. *Space Time Play - Computer Games, Architecture and Urbanism: The Next Level*. Basel / Boston / Berlin: Birkhäuser Verlag, 2007.
- Branigan, Edward. *Narrative Comprehension and Film*. London / New York: Routledge, 1992.
- Brecht, Bertold. *Brecht on Theatre*. Trans. John Willet: Hill and Wang, 1982.

- Brook, Peter. *The Empty Space*. (1968). New York: Atheneum, 1984.
- Buckles, Mary Ann. "Interactive Fiction: The Computer Storygame *Adventure*." University of California San Diego, 1985.
- . "Interactive Fiction as Literature: Adventure Games Have a Literary Lineage." *Byte*. May (1987): 135-42.
- Caillois, Roger. *Man, Play and Games*. 1958. Trans. Meyer Barash. Paperback ed. Urbana and Chicago: University of Illinois Press, 1961.
- Campbell, Joseph. *The Hero of a Thousand Faces*. Princeton University Press, 1972.
- Carroll, Lewis. *Lewis Carroll's Games and Puzzles*. Ed. Edward Wakeling. New York: Dover Publications, 1992.
- . *Alice's Adventures in Wonderland / through the Looking-Glass*. (1865, 1872). Penguin Classics. Ed. Hugh Haughton. London: Penguin Books, 1998.
- Carson, Don. "Environmental Storytelling: Creating Immersive 3D Worlds Using Lessons Learned from the Theme Park Industry." *Gamasutra* (2000).  
<[http://www.gamasutra.com/view/feature/3186/environmental\\_storytelling\\_.php](http://www.gamasutra.com/view/feature/3186/environmental_storytelling_.php)> (accessed November 8, 2009).
- Certeau, Michel de. *The Practice of Everyday Life*. Berkeley: University of California Press, 1984.
- Chatman, Seymour. *Story and Discourse: Narrative Structure in Fiction and Film*. Ithaca: Cornell University Press, 1980.
- Church, Doug. "Formal Abstract Design Tools." *The Game Design Reader: A Rules of Play Anthology*. Eds. Katie Salen and Eric Zimmerman. Cambridge, MA: MIT Press, 2006.
- Citizen Kane*. Dir. Welles, Orson. Film. Orson Welles. 1941.

- Cook, David “Zeb”, et al. *Planescape Campaign Setting (Advanced Dungeons and Dragons)*. Wizards of the Coast, 1994.
- Costikyan, Greg. “I Have No Words & I Must Design.” *The Game Design Reader : A Rules of Play Anthology*. Eds. Katie Salen and Eric Zimmerman. Cambridge, MA: MIT Press, 2006.
- . “Games, Storytelling and Breaking the String.” *Second Person: Role-Playing and Story in Games and Playable Media*. Eds. Pat Harrigan and Noah Wardrip-Fruin. Cambridge, MA / London, England: The MIT Press, 2007.
- Crawford, Chris. *Chris Crawford on Game Design*. New Riders Games. Indianapolis, IN: New Riders, 2005.
- . *Chris Crawford on Interactive Storytelling*. New Riders Games. 1st ed. Berkeley, Ca: New Riders, 2005.
- Danesi, Marcel. *The Puzzle Instinct*. Bloomington, IN: Indiana University Press, 2002.
- Eco, Umberto. *Semiotics and the Philosophy of Language*. Bloomington: Indiana University Press, 1984.
- Egri, Lajos. *The Art of Dramatic Writing*. Simon and Schuster, 1960.
- Eisner, Will. *Comics and Sequential Art*. Porthouse Press, 2001.
- Eskelinen, Markku. “The Gaming Situation.” *Game Studies* 1.1 (2001). 16 September 2009 <<http://www.gamestudies.org/0101/eskelinen/>> (accessed November 8, 2009).
- . “Towards Computer Game Studies.” *First Person: New Media as Story, Performance, and Game*. Cambridge, MA.: MIT Press, 2004. xiii, 331 p.
- Evers, M. “The Boys with the Flying Fingers: South Korea Turns Pc Gaming into a Spectator Sport.” *Spiegel Online* (2006). <<http://www.spiegel.de/international/spiegel/0,1518,399476,00.html>> (accessed

November 8, 2009).

Fernández Vara, Clara. "Labyrinth and Maze: Video Game Navigation Challenges." *Space Time Play - Computer Games, Architecture and Urbanism: The Next Level*. Eds. Friedrich von Borries, Steffen P. Walz and Matthias Bottger. Basel: Birkhäuser Verlag, 2007.

---. "Shaping Player Experience in Adventure Games: History of the Adventure Game Interface." *Extending Experiences*. Eds. Amyris Fernandez, Olli Leino and Hana Wirman. Rovaniemi: Lapland University Press, 2008.

Fernández Vara, Clara, Jose P. Zagal, and Michael Mateas. "Evolution of Spatial Configurations in Videogames." *Worlds in Play: International Perspectives on Digital Games Research*. Eds. Suzanne De Castell and Jennifer Jenson. New York: Peter Lang, 2007. 159-68.

Firth, Roger, Sonja Kesserich, and Dennis G. Jerz. *The Inform Beginner's Guide*. Placet, 2003.

Fitzgerald, F. Scott. *The Notebooks of F. Scott Fitzgerald*. Ed. Matthew J. Bruccoli. New York: Harcourt Brace Jovanovich / Bruccoli Clark, 1978.

Forster, E.M. *Aspects of the Novel*. (1927). San Diego: Harcourt, 1985.

Frasca, Gonzalo. "Videogames of the Oppressed: Critical Thinking, Education Tolerance, and Other Trivial Issues." *First Person: New Media as Story, Performance, and Game*. Eds. Noah Wardrip-Fruin and Pat Harrigan. Cambridge, MA: MIT Press, 2004.

---. "Ludology Meets Narratology: Similitude and Differences between (Video)Games and Narrative." *Parnasso.3* (1999).

---. "Videogames of the Oppressed." Georgia Institute of Technology, 2001.

---. "Simulation 101: Simulation Versus Representation". 2001.  
<<http://www.ludology.org/articles/sim1/simulation101.html>> (accessed November 8, 2009).

---. "Ludologists Love Stories Too: Notes from a Debate That Never Took Place." *Level Up* Ed.



Marinka Copier and Joost Raessens.: University of Utrecht.

Fullerton, Tracy, Christopher Swain, and Steven Hoffman. *Game Design Workshop: Designing, Prototyping and Playtesting Games*. Berkeley, CA: CMP Books, 2004.

Gabler, Kyle, et al. "How to Prototype a Game in under 7 Days: Tips and Tricks from 4 Grad Students Who Made over 50 Games in 1 Semester." *Gamasutra* (2005).  
<[http://www.gamasutra.com/features/20051026/gabler\\_01.shtml](http://www.gamasutra.com/features/20051026/gabler_01.shtml)> (accessed November 8, 2009).

Garner, Stanton B. *Bodied Spaces: Phenomenology and Performance in Contemporary Drama*. 1994. Cornell University Press, 1994.

Gee, James Paul. *What Video Games Have to Teach Us About Learning and Literacy*. 1st ed. New York: Palgrave Macmillan, 2003.

Genette, Gérard. *Narrative Discourse: An Essay in Method*. Ithaca, N.Y.: Cornell University Press, 1980.

Gingold, Chaim. "Miniature Gardens & Magic Crayons: Games, Spaces, & Worlds." Georgia Institute of Technology 2003.

Goffman, Erving. *The Presentation of Self in Everyday Life*. University of Edinburgh. Social Sciences Research Centre; Monograph No. 2. Edinburgh,: University of Edinburgh, Social Sciences Research Centre, 1956.

---. *Frame Analysis: An Essay on the Organization of Experience*. Cambridge: Harvard University Press, 1974.

Gygax, Gary, and Dave Arneson. *Dungeons and Dragons*. Tactical Studies Rules (TSR), 1974.

Harrigan, Pat, and Noah Wardrip-Fruin. *Second Person: Role-Playing and Story in Games and Playable Media*. Cambridge, MA: The MIT Press, 2007.

Henderson, Diana. "A Shrew for the Times." *Shakespeare: The Movie: Popularizing the Plays*

- on Film, Tv and Video*. Eds. Linda E. Boose and Richard Burt. New York / London: Routledge, 1997.
- Hovanec, Helene. *The Puzzler's Paradise - From the Garden of Eden to the Computer Age*. Paddington Press Ltd., 1978.
- Howard, Jeff. *Quests: Design, Theory, and History in Games and Narratives*. Wellesley, MA: A K Peters, 2008.
- Huizinga, Johan. *Homo Ludens; a Study of the Play-Element in Culture*. Humanitas, Beacon Reprints in Humanities. Boston: Beacon Press, 1955.
- Hunicke, Robin, Marc LeBlanc, and Robert Zubek. "MDA: A Formal Approach to Game Design and Game Research." *AAAI-04 Workshop on Challenges in Game AI*.
- Isbister, Katherine. *Better Game Characters by Design: A Psychological Approach*. San Francisco, CA: Morgan Kaufmann, 2006.
- Jakobson, R., Fant G., and Morris Halle. *Preliminaries to Speech Analysis: The Distinctive Features and Their Correlates*. Cambridge, MA: MIT Press, 1952.
- Jenkins, Henry. "Game Design as Narrative Architecture." *First Person: New Media as Story, Performance, and Game*. Eds. Noah Wardrip-Fruin and Pat Harrigan. Cambridge, MA.: MIT Press, 2004. 118-30.
- . *Convergence Culture: Where Old and New Media Collide*. New York: NYU Press, 2006.
- Järvinen, Aki. "Games without Frontiers: Theories and Methods for Game Studies and Design." University of Tampere, Finland, 2008.
- Jerz, Dennis G. "Somewhere Nearby Is Colossal Cave: Examining Will Crowther's Original 'Adventure' In Code and in Kentucky." *Digital Humanities Quarterly* 1.2 (2007). <<http://digitalhumanities.org/dhq/vol/001/2/0000009.html>> (accessed November 8, 2009).
- Juul, Jesper. "Games Telling Stories? A Brief Note on Games and Narratives." *Game Studies* 1.1

(2001).

---. *Half-Real: Video Games between Real Rules and Fictional Worlds*. Cambridge, MA: MIT Press, 2005.

---. "A Certain Level of Abstraction." *DiGRA 2007: Situated Play*. Ed. Akira Baba.

Kern, Hermann. *Through the Labyrinth: Designs and Meanings over 5,000 Years*. Munich: Prestel, 2000.

Kim, Scott. "What Is a Puzzle?" *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*. Eds. Tracy Fullerton, Christopher Swain and Steven Hoffman. 2nd ed. Burlington, MA: Morgan Kaufmann, 2008.

Klabbers, Jan H. G. *The Magic Circle: Principles of Gaming and Simulation*. 2nd. ed. Rotterdam / Taipei: Sense Publishers, 2008.

Klastrup, Lisbeth. "A Poetics of Virtual Worlds." *Digital Arts and Culture*.

Klevjer, Rune. "In Defense of Cutscenes." *Computer Games and Digital Cultures Conference*. Ed. Frans Mäyra: Tampere University Press.

---. "What Is the Avatar? Fiction and Embodiment in Avatar-Based Singleplayer Computer Games." University of Bergen, 2006.

Krawczyk, Marianne, and Jeannie Novak. *Game Development Essentials: Game Story & Character Development*. Delmar Cengage Learning, 2006.

Lankoski, Petri, and Staffan Bjork. "Character-Driven Game Design: Characters, Conflicts and Gameplay." *GDTW, Sixth International Conference in Game Design and Technology*.

Laurel, Brenda. *Computers as Theatre*. Reading, MA: Addison-Wesley Publishing Company, 1993.

- LeBlanc, Marc. "Tools for Creating Dramatic Game Dynamics." *The Game Design Reader: A Rules of Play Anthology*. Eds. Katie Salen and Eric Zimmerman. Cambridge, MA: MIT Press, 2006.
- Lebling, P. David, Marc S. Blank, and Timothy A. Anderson. "Zork: A Computerized Fantasy Simulation Game." *IEEE Computer* 12.4 (1979): 51-59.
- Linde, Charlotte, and William Labov. "Spatial Networks as a Site for the Study of Language and Thought." *Language* 51.4 (1975): 924-39.
- Lynch, Kevin. *The Image of the City*. Cambridge, MA: The Technology Press & Harvard University Press, 1960.
- MacDonald, Laura. "Quantic Dream - David Cage and Guillaume De Fondaumiere Interview." *Adventure Gamers* (2005). <<http://www.adventuregamers.com/article/id,554>> (accessed November 8, 2009).
- Mackay, Daniel. *The Fantasy Role-Playing Game*. Jefferson: McFarland & Company, 2001.
- Mateas, Michael. "A Preliminary Poetics for Interactive Drama and Games." *First Person: New Media as Story, Performance, and Game*. Eds. Noah Wardrip-Fruin and Pat Harrigan. Cambridge, MA: MIT Press, 2004.
- Mateas, Michael, and Andrew Stern. "Interaction and Narrative." *The Game Design Reader: A Rules of Play Anthology*. Eds. Katie Salen and Eric Zimmerman. Cambridge, MA: MIT Press, 2006.
- McGonigal, Jane. "A Real Little Game: The Performance of Belief in Pervasive Play." *Level Up*.
- Mcquillan, Martin, ed. *The Narrative Reader*. New York: Routledge, 2000.
- Meadows, Mark Stephen. *Pause & Effect: The Art of Interactive Narrative*. Indianapolis, IN: New Riders, 2003.
- Meretzky, Steve. "Building Character: An Analysis of Character Creation." *Gamasutra* (2001).

<[http://www.gamasutra.com/resource\\_guide/20011119/meretzky\\_03.htm](http://www.gamasutra.com/resource_guide/20011119/meretzky_03.htm)> (accessed November 8, 2009).

---. "The Creation of Floyd the Robot in Planetfall." *Second Person: Role-Playing and Story in Games and Playable Media*. Eds. Pat Harrigan and Noah Wardrip-Fruin. Cambridge, MA: MIT Press, 2007.

Montfort, Nick. *Twisty Little Passages: An Approach to Interactive Fiction*. Cambridge, MA: MIT Press, 2003.

---. "Fretting the Player Character." *Second Person: Role-Playing and Story in Games and Playable Media*. Eds. Pat Harrigan and Noah Wardrip-Fruin. Cambridge, MA: MIT Press, 2007.

Murray, Janet. "Building Coherent Plots in Interactive Fiction." *IEEE Intelligent Systems and Their Applications* 13.6 (1998): 18-21.

---. *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. Cambridge, MA: The MIT Press, 2001.

---. "From Game-Story to Cyberdrama." *First Person: New Media as Story, Performance, and Game*. Cambridge, MA: MIT Press, 2004. xiii, 331 p.

Myers, David. "The Attack of the Backstories (and Why They Won't Win)." *Level Up Conference Proceedings*. Ed. Marinka Copier and Joost Raessens: University of Utrecht.

Nelson, Graham. "The Craft of Adventure: Five Articles on the Design of Adventure Games (Second Edition)." (1995). <<http://mirror.ifarchive.org/if-archive/info/Craft.Of.Adventure.pdf>> (accessed November 8, 2009).

---. *The Inform Designer's Manual*. Ed. Gareth Rees: Interactive Fiction Library, 2001.

Nitsche, Michael. *Video Game Spaces: Image, Play, and Structure in 3D Worlds*. Cambridge, MA: MIT Press, 2009.

---. *Mapping Time in Video Games*. DiGRA: Situated Play. 2007.

Novak, Marcos. "Liquid Architectures in Cyberspace." *Multimedia - from Wagner to Virtual Reality*. Eds. R. Packer and K. Jordan. vols: W. W. Norton, 2001.

Partlett, David. *The Oxford History of Board Games*. Oxford University Press, 1999.

Pavis, Patrice. *Theatre at the Crossroads of Culture*. London: Routledge, 1991.

Pearce, Celia. "Story as Play Space: Narrative in Games." *Game On*. Eds. Lucian King and Conrad Bain. London: Barbican, 2002.

---. "Sims, Battlebots, Cellular Automata God and Go: A Conversation with Will Wright." *Game Studies* 2.1 (2002). <<http://gamestudies.org/0102/pearce/>> (accessed November 8, 2009).

---. "Towards a Game Theory of Game." *First Person: New Media as Story, Performance, and Game*. Eds. Noah Wardrip-Fruin and Pat Harrigan. Cambridge, MA: MIT Press, 2004. xiii, 331 p.

---. "Narrative Environments: From Disneyland to World of Warcraft." *Space Time Play - Computer Games, Architecture and Urbanism: The Next Level*. Eds. Friedrich von Borries, Steffen P. Walz and Matthias Bottger. Boston / Berlin: Birkhäuser Verlag, 2007.

Peirce, Charles. *Essential Peirce: Selected Philosophical Writings*. Ed. Christian Knowsel Nathan Houser. Vol. 2. Bloomington, IN: Indiana University Press, 1998.

Poe, Edgar Allan. *Edgar Allan Poe: Fiction and Poetry, Complete and Unabridged*. Ed. Dawn B. Sova. New York: Barnes and Noble, 2006.

Prince, Gerald. *Dictionary of Narratology (Revised Edition)*. Lincoln & London: University of Nebraska Press, 2003.

Propp, V. I. Fe A. Fe. *Morphology of the Folktale*. 1930. 2d ed. Austin,: University of Texas Press, 1968.

- Rau, Anja. "Reload - Yes / No: Clashing Times in Graphic Adventure Games." *Computer Games & Digital Textualities*.
- Remo, Chris. "The Last Express: Revisiting an Unsung Classic." *Gamasutra* (2008).  
<[http://www.gamasutra.com/view/feature/3862/the\\_last\\_express\\_revisiting\\_an\\_.php](http://www.gamasutra.com/view/feature/3862/the_last_express_revisiting_an_.php)>  
(accessed November 8, 2009).
- Roberts, Michael. "Choosing a Conversation System". 2001-2007.  
<<http://www.tads.org/howto/convbkg.htm>> (accessed November 8, 2009).
- Rollings, Andrew, and Ernest Adams. *Andrew Rollings and Ernest Adams on Game Design*. Indianapolis, IN: New Riders Publishing, 2003.
- Ryan, Marie-Laure. "Beyond Myth and Metaphor: The Case of Narrative in Digital Media." *Game Studies* 1.1 (2001).
- . *Narrative as Virtual Reality*. Baltimore: The Johns Hopkins University Press, 2001.
- . "Beyond Myth and Metaphor: Narrative in Digital Media." *Poetics Today* 23.4 (2002): 581-609.
- Salen, Katie, and Eric Zimmerman. *Rules of Play : Game Design Fundamentals*. Cambridge, MA: MIT Press, 2004.
- . *The Game Design Reader: A Rules of Play Anthology*. Cambridge, MA: MIT Press, 2006.
- Saltz, David. "The Art of Interaction: Interactivity, Performativity, and Computers." *Journal of Aesthetics and Art Criticism* 55.2 (1997): 117-27.
- Saussure, Ferdinand de, et al. *Course in General Linguistics*. London: Duckworth, 1983.
- Schechner, Richard. *Between Theater and Anthropology*. Philadelphia: University of Pennsylvania Press, 1985.

---. *Performance Theory*. London: Routledge, 2003.

---. *Performance Studies: An Introduction*. 2nd ed. New York: Routledge, 2006.

Shakespeare, William. *A Midsummer Night's Dream*. The Oxford Shakespeare. Ed. Peter Holland. Oxford / New York: Oxford University Press, 1994.

Shakespeare, William. *King Henry V*. The Arden Shakespeare. Eds. Richard Proudfoot, Ann Thompson and David Scott Kastan: Routledge, 1997.

Shneiderman, Ben. "Direct Manipulation: A Step Beyond Programming Languages." *The New Media Reader*. Eds. Noah Wardrip-Fruin and Nick Montfort. Cambridge, MA: MIT Press, 2003. 486-99.

Shank, Roger C., and Robert P. Abelson. *Scripts, Plans, Goals and Understanding*. Hillsdale, NJ: Lawrence Erlbaum, 1977.

Short, Emily. "NPC Characterization". <<http://emshort.home.mindspring.com/NPC4.htm>> (accessed November 8, 2009).

Sicart, Miguel. "Defining Game Mechanics." *Game Studies* 8.2 (2008). <<http://gamestudies.org/0802/articles/sicart>> (accessed November 8, 2009).

Smullyan, Raymond. *The Lady or the Tiger? And Other Logic Puzzles*. New York: Times Books, 1982.

Spector, Warren. "Next-Generation Storytelling." *The Escapist*.94 (2007).

Squire, Kurt, and Henry Jenkins. "The Art of Contested Spaces." *Game On*. Eds. Lucian King and Conrad Bain. London: Barbican, 2002.

Sternberg, Robert J. *Beyond IQ: A Triarchic Theory of Human Intelligence*. Cambridge: Cambridge University Press, 1985.



- Sturken, M., and L. Cartwright. *Practices of Looking: An Introduction to Visual Culture*. Oxford: Oxford University Press, 2001.
- Suits, Bernard. "The Detective Story: A Case Study of Games in Literature." *Canadian Review of Comparative Literature* 12.2 (1985): 200-19.
- Tolkien, J.R.R. *The Lord of the Rings*. (1966). HarperCollins, 2001.
- Turkle, Sherry. "Videogames and the Computer Holding Power." *The New Media Reader*. Eds. Noah Wardrip-Fruin and Nick Montfort. Cambridge, MA: MIT Press, 2003.
- Turner, Victor Witter. *The Ritual Process: Structure and Anti-Structure*. Symbol, Myth and Ritual. Ed. Victor Witter Turner. Ithaca, NY: Cornell University Press, 1969.
- Wardrip-Fruin, Noah, and Pat Harrigan. *First Person: New Media as Story, Performance, and Game*. Cambridge, MA: MIT Press, 2004.
- Williams, Roberta. "Winning Strategies for Adventures." *The Online Letter* 1.1 (1981): 7.
- Wolpaw, Eric. "Death of Adventure Games." *Old Man Murray* (2000).  
<<http://www.oldmanmurray.com/features/77.html>> (accessed November 8, 2009).
- Wood, John. "Player Character Identity in If." *XYZZY News*.9 (1996): 7-11.
- Zagal, Jose P. "Game Ontology". <[http://www.gameontology.org/index.php/Main\\_Page](http://www.gameontology.org/index.php/Main_Page)> (accessed November 8, 2009).
- Zagal, Jose P., et al. "Towards an Ontological Language for Game Analysis." *DiGRA 2005*.
- Zagal, Jose P., Clara Fernández Vara, and Michael Mateas. "Rounds, Levels and Waves: The Early Evolution of Gameplay Segmentation." *Games and Culture* 3.2 (2008): 175-98.
- Zagal, Jose P., and Michael Mateas. "Temporal Frames: A Unifying Framework for the Analysis

of Game Temporality.” *DiGRA: Situated Play*.

## GAMES AND OTHER SOFTWARE CITED

Trilobyte. *The 7th Guest*. Virgin Games, 1997. Windows.

---. *The 11th Hour*. Virgin Interactive Entertainment, 1995. Windows.

Crowther, Bill, and Don Woods. *Adventure*, 1975-7. PDP-10.

Barlow, Sam. *Aisle*, 1999. Z-Code.

Cognetics Corporation. *Amnesia*. Electronic Arts, 1986. Commodore 64.

AWE Production. *And Then There Were None*. The Adventure Company, 2008. Wii.

Nintendo EAD. *Animal Crossing*. Nintendo of America, 2002. GameCube.

Capcom. *Apollo Justice: Ace Attorney*. Capcom Entertainment, 2008. Nintendo DS.

Infocom. *Arthur: The Quest for Excalibur*. Infocom, 1989. DOS.

Atari. *Asteroids*, 1979. Arcade.

Pulse Entertainment. *Bad Mojo*. Acclaim Entertainment, 1996. Windows.

Infocom. *Ballyhoo*. Infocom, 1986. DOS.

PopCap Games. *Bejewelled*. PopCap Games, 2000. Windows.

Infocom. *Beyond Zork: The Coconut of Quendor*. Infocom, 1987. DOS.

2K Boston, and 2K Australia. *Bioshock*. 2K Games, 2007. XBOX 360.

Channel 4. *Bow Street Runners*, 2008. <http://www.channel4.com/history/microsites/C/city-of-vice/game/bow-street-runner/game.html>.

Atari. *Breakout*. Atari, 1978. Atari 2600.

Revolution Software. *Broken Sword: The Shadow of the Templars*. Virgin Interactive Entertainment, 1996. DOS.

Infocom. *Bureaucracy*. Infocom, 1987. DOS.

Konami of America. *Castlevania: Symphony of the Night*. Konami Computer Entertainment Tokyo, 1997. Playstation.

Realmforge Studios GmbH. *Ceville*. Kalypso Media GmbH, 2009. Windows.

Office Create. *Cooking Mama*. Majesco Entertainment Company, 2006. Nintendo DS.

Takagi, Toshimitsu. *Crimson Room*. Fasco-CS, 2004. Flash. [http://www.fasco-csc.com/works/crimson/crimson\\_e.php](http://www.fasco-csc.com/works/crimson/crimson_e.php).

Entertainment Hawaii, Konami Computer. *Dance Dance Revolution Ultramix 3*. Konami Digital Entertainment, 2005. XBOX.

Lucasarts Entertainment Company. *Day of the Tentacle*. Lucasarts Entertainment Company, 1993. DOS.

Infocom. *Deadline*. Infocom, 1984. DOS.

Rainbow Studios. *Disney / Pixar Cars*. THQ, 2006. Wii.

Bethesda Game Studios. *The Elder Scrolls: Oblivion*. 2K Games, 2007. XBOX 360.

Weizembaum, Joseph. *Eliza*, 1964-6. SLIP.

Legend Entertainment Company. *Eric the Unready*. Legend Entertainment Company, 1993. DOS.

Procedural Arts LLC. *Façade*. Procedural Arts LLC, 2005. Windows.

Quantic Dream. *Fahrenheit A.K.A. Indigo Prophecy*. Atari, 2005. Windows.

Square. *Final Fantasy VII*. Sony Computer Entertainment of America, 1997. 1997.

Lucasarts Entertainment Company. *Full Throttle*. Lucasarts Entertainment Company, 1995. Windows.

Sierra On-Line. *Gabriel Knight 3: Blood of the Sacred*. Sierra On-Line, 1999. 1999.

---. *Gabriel Knight: Sins of the Fathers*. Sierra On Line, 1993. DOS.

Short, Emily. *Galatea*, 2000. Z-code.

Namco. *Galaxian*, 1979. Arcade.

Rockstar North. *Grand Theft Auto: Vice City*. Rockstar Games, 2002. Playstation.

Lucasarts Entertainment Company. *Grim Fandango*. Lucasarts Entertainment Company, 1998. Windows.

Pacian, C.E.J. *Gun Mute*, 2008. TADS 3.

Valve. *Half-Life 2*. Vivendi Universal Games, 2004. Windows.

Electronic Arts UK, and Visual Sciences. *Harry Potter: Quidditch World Cup*. Electronic Arts,

2003. GameCube.

Infocom. *The Hitchhiker's Guide to the Galaxy*. Infocom, 1984. DOS.

Beam Software. *The Hobbit*. Melbourne House, 1983. Commodore 64.

CING. *Hotel Dusk: Room 215*. Nintendo of America, 2007. Nintendo DS.

Bergman, Jason. *If Quake* 2004. Z-Code.

Lucasfilm Games. *Indiana Jones and the Fate of Atlantis*. Lucasfilm Games, 1992. DOS.

---. *Indiana Jones and the Last Crusade: The Graphic Adventure*. Lucasfilm Games, 1989. DOS.

Sierra On-Line. *King's Quest Iv: The Perils of Rosella*. Sierra On Line, 1988. DOS.

---. *King's Quest V: Absence Makes the Heart Go Yonder!* Sierra On Line, 1988. DOS.

---. *King's Quest: Quest for the Crown*. Sierra On-Line, 1987. DOS.

Smoking Car Productions. *The Last Express*. Brøderbund Software, 1997. Macintosh.

Infocom. *Leather Goddesses of Phobos*. Infocom, 1986. DOS.

Nintendo EAD. *The Legend of Zelda: Majora's Mask*. Nintendo of America, 2000. 2000.

---. *The Legend of Zelda: Phantom Hourglass*. Nintendo of America, 2007. Nintendo DS.

Funcom Oslo. *The Longest Journey*. Funcom Oslo, 2000. Windows.

Lucasfilm Games. *Loom*. Lucasfilm Games, 1990. DOS.

Turbine. *Lord of the Rings Online: Shadows of Angmar*. Midway Games, 2007. Windows.

Mythos Software. *The Lost Files of Sherlock Holmes: The Case of the Serrated Scalpel*. Electronic Arts, 1992. DOS.

Infocom. *The Lurking Horror*. Infocom, 1987. DOS.

NevoSoft. *Magic Academy*. NevoSoft, 2007. 2007.

Lucasfilm Games. *Maniac Mansion*. Lucasfilm Games, 1987. 1987.

Atari. *Marble Madness*, 1984. Arcade.

Nintendo EAD. *Mario Kart: Double Dash*. Nintendo of America, 2003. GameCube.

Infocom. *A Mind Forever Voyaging*. Infocom, 1985. DOS.

Lucasfilm Games. *Monkey Island 2: Lechuck's Revenge*. Lucasfilm Games, 1991. DOS.

Darrow, Charles. *Monopoly*. Parker Brothers, 1935. Board Game.

Infocom. *Moonmist*. Infocom, 1986. DOS.

Cyan Worlds, and Red Orb Entertainment. *Myst (Masterpiece Edition)*. Mindscape, 1999. Windows.

Presto Studios. *Myst Iii: Exile*. Ubi Soft Entertainment, 2001. Windows.

On-Line Systems. *Mystery House*. On-Line Systems, 1980. Apple II.

Albartus, Jan. *Mystery of Time and Space*, 2001-9. <http://motas.gamesplanet.be/home/>.

Capcom. *Phoenix Wright: Ace Attorney*. Capcom Entertainment, 2005. Nintendo DS.

Black Isle Studios. *Planescape: Torment*. Interplay Entertainment, 1999. Windows.

Infocom. *Planetfall*. Infocom, 1983. DOS.

---. *Plundered Hearts*. Infocom, 1987. DOS.

Atari. *Pong*, 1972. Arcade.

Level-5. *Professor Layton and the Curious Village*. Nintendo of America, 2008. Nintendo of America.

Productions, Double Fine. *Psychonauts*. Majesco Entertainment Company, 2005. XBOX.

Harmonix Music Systems. *Rock Band*. MTV Games, 2007. XBOX 360.

Anderson, Jens. *Rorschach*, 2007. PC.

Sagrario, Valentin. *Sagrario's Room Escape*, 2008. <http://sagrarios.room.escape.fizzlebot.com/>.

Lucasarts Entertainment Company. *Sam & Max Hit the Road*. Lucasarts Entertainment Company, 1993. DOS.

Telltale. *Sam & Max Save the World: Season One (Culture Shock)*. The Adventure Company, 2007. Windows.

Lucasfilm Games. *The Secret of Monkey Island*. Lucasfilm Games, 1990. DOS.

---. *Secret Weapons of the Luftwaffe*. Lucasfilm Entertainment Company, 1990. DOS.



Frogwares Game Development Studio. *Sherlock Holmes: The Awakened*. CDV Software Entertainment USA, 2007. Windows.

Cadre, Adam. *Shrapnel*, 2000. Z-Code.

Games, Firaxis. *Sid Meier's Civilization Iii*. MacSoft, 2002. Macintosh.

Maxis Software. *The Sims*. Electronic Arts, 2000. Windows.

---. *The Sims 2*. Electronic Arts, 2004. Windows.

Sierra On-Line. *Space Quest I: Roger Wilco in the Sarien Encounter*. Sierra On-Line, 1991. DOS.

---. *Space Quest: The Sarien Encounter*. Sierra On-Line, 1986. DOS.

BioWare Corporation. *Star Wars: Knights of the Old Republic*. LucasArts, 2003. Windows.

Lucasarts Entertainment Company. *Star Wars: X-Wing*. Lucasarts Entertainment Company, 1993. DOS.

Blizzard Entertainment. *Starcraft*. Blizzard Entertainment, 1998. Windows.

Capcom. *Street Fighter II*. U.S. Gold, 1992. DOS.

Atari. *Super Breakout*. Atari, 1981. Atari 2600.

Nintendo EAD. *Super Mario 64*. Nintendo of America, 1996. Nintendo 64.

Infocom. *Suspended: A Cryogenic Nightmare*. Infocom, 1982. DOS.

Microïds. *Syberia*. The Adventure Company, 2002. Windows.

Telltale. *Tales of Monkey Island: The Launch of the Screaming Narwhal*. Telltale, 2009. Windows.

Ravipinto, Daniel. *Tapestry*, 1996. Z-Code.

AcademySoft. *Tetris*, 1986. DOS.

CING. *Trace Memory / Another Code*. Nintendo of America, 2005. Nintendo DS.

Atlus. *Trauma Center: Under the Knife*. Atlus U.S.A., 2005. Nintendo DS.

Telarium. *Treasure Island*. Telarium, 1985. MSX 2.

Infocom. *Trinity*. Infocom, 1986. DOS.

ORIGIN Systems. *Ultima Online*. Electronic Arts, 1997. Windows.

Hitmaker. *Virtua Tennis*. SEGA of America, 2000. Dreamcast.

Psygnosis Limited. *Wipeout*. Sony Computer Entertainment of America, 1995. Playstation.

Blizzard Entertainment. *World of Warcraft*. Blizzard Entertainment, 2004. Windows.

Infocom. *Zork: The Great Underground Empire*. Infocom, 1980. DOS.