

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

**CHANGEFUL TALES:
DESIGN-DRIVEN APPROACHES TOWARD
MORE EXPRESSIVE STORYGAMES**

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE

by

Aaron A. Reed

June 2017

The Dissertation of Aaron A. Reed
is approved:

Noah Wardrip-Fruin, Chair

Michael Mateas

Michael Chemers

Dean Tyrus Miller
Vice Provost and Dean of Graduate Studies

Copyright © by

Aaron A. Reed

2017

Table of Contents

List of Figures	viii
List of Tables	xii
Abstract	xiii
Acknowledgments	xv
Introduction	1
1 Framework	15
1.1 Vocabulary	15
1.1.1 Foundational terms	15
1.1.2 Storygames	18
1.1.2.1 <i>Adventure</i> as prototypical storygame	19
1.1.2.2 What Isn't a Storygame?	21
1.1.3 Expressive Input	24
1.1.4 Why Fiction?	27
1.2 A Framework for Storygame Discussion	30
1.2.1 The Slipperiness of Genre	30
1.2.2 Inputs, Events, and Actions	31
1.2.3 Mechanics and Dynamics	32
1.2.4 Operational Logics	33
1.2.5 Narrative Mechanics	34
1.2.6 Narrative Logics	36
1.2.7 The Choice Graph: A Standard Narrative Logic	38
2 The Adventure Game: An Existing Storygame Mode	44
2.1 Definition	46
2.2 Eureka Stories	56
2.3 The Adventure Triangle and its Flaws	60
2.3.1 Instability	65

2.4	<i>Blue Lacuna</i>	66
2.5	Three Design Solutions	69
2.5.1	<i>The Witness</i>	70
2.5.2	<i>Firewatch</i>	78
2.5.3	<i>Her Story</i>	86
2.6	A Technological Fix?	94
3	Sculptural Fiction	98
3.1	Sculptural fiction overview	102
3.2	Connections to existing concepts and works	107
3.2.1	Pre-digital works	107
3.2.2	Hypertext Theory	113
3.2.3	Other Theoretical Work	115
3.2.4	Released Games and Systems	117
3.2.4.1	StoryNexus	117
3.2.4.2	<i>King of Dragon Pass</i>	121
3.2.4.3	<i>Storyteller</i>	122
3.3	Early Sculptural Fiction Projects	124
3.3.1	<i>Perfect</i>	124
3.3.2	<i>18 Cadence</i>	126
3.3.3	<i>Hollywood Visionary</i>	130
3.4	Challenges with Sculptural Fiction	133
3.4.1	Problems for Authors	134
3.4.1.1	Understanding the Authoring Space	134
3.4.1.2	Finding bad states through random testing	137
3.4.2	Problems for Players	140
3.4.2.1	Orientation	140
3.4.2.2	Fires in the Desert	142
3.4.2.3	Loss of Immersion	144
3.5	Futures for Sculptural Fiction	146
3.5.1	Supportive Play Partners	147
3.5.2	Flexible Nodes	148
4	Sculptural Fiction Case Study: <i>The Ice-Bound Concordance</i>	152
4.1	Overview	152
4.2	Narrative Engine	154
4.2.1	Symbol Selection	156
4.2.2	Player Interaction	158
4.2.3	Viewing expanded lexia text	160
4.2.4	Template Language	163
4.2.5	Resolving Endings	166
4.3	Conversations with KRIS	168
4.4	Revelations	169

4.5	Authoring for <i>Ice-Bound</i>	172
4.6	<i>Ice-Bound</i> as Sculptural Fiction	176
5	Social Simulation	178
5.1	Introduction	178
5.2	Defining Social Simulation	185
5.3	Survey of Major Social Simulation Storygames	190
5.3.1	<i>Prom Week</i>	191
5.3.1.1	Game Overview	191
5.3.1.2	Core Gameplay Loop	192
5.3.1.3	Larger Game Structure	194
5.3.1.4	<i>Prom Week</i> Analysis	196
5.3.2	<i>Blood & Laurels</i>	199
5.3.3	<i>Redshirt</i>	207
5.3.4	Other Works	212
5.3.4.1	<i>Façade</i>	212
5.3.4.2	StoryBricks	213
5.3.4.3	<i>Siboot</i> and other Chris Crawford projects	215
5.3.4.4	<i>Black Closet</i> and Hanako Games	216
5.3.4.5	Non-narrative social simulations	218
5.4	Characterizing Social Simulation Storygames	219
5.4.1	Schema, Rules, and Narrations: Three Types of Content	221
5.4.2	Multiplicative, Not Additive Complexity	224
5.4.3	Stories About Specific Systems, not Specific People	224
5.4.4	Breaking out of Ruts	228
5.5	Challenges with Social Simulation Games	230
5.5.1	Prototyping and Design	231
5.5.2	Simulation Fidelity	234
5.5.3	Making the System Visible	239
5.5.4	Authoring and Scoping	241
5.5.5	Testing	245
5.5.6	Rule Sorting, Precedence and Weighting	246
5.5.7	Reuse of Content	250
5.5.8	The Bad Parts of Emergent Surprise	252
5.6	Futures for Social Simulation	257
6	Social Simulation Case Study	263
6.1	Authoring for <i>Prom Week</i>	264
6.1.1	Overview	264
6.1.2	Templates and History	270
6.1.3	Other Writing	273
6.1.4	Surprises	276
6.2	Ensemble: A Design Tool	281

7	Collaborative Storygames	286
7.1	Definition	290
7.1.1	Connection to tabletop roleplaying theory	295
7.1.2	Connection to existing collaborative fictions design work	297
7.2	Design Framework	299
7.2.1	Simulative Spectrum: World to Story	299
7.2.2	Performative Spectrum: Author to Improviser	301
7.2.3	Game Master as mediator between two	302
7.2.4	Four Roles	304
7.2.4.1	Generation	304
7.2.4.2	Storywrighting	305
7.2.4.3	Negotiation	306
7.2.4.4	Administration	307
7.2.4.5	(Performance?)	308
7.3	Example Analog Games	308
7.3.1	Andrew Plotkin’s Journal Writing Game	308
7.3.2	<i>Microscope</i>	311
7.3.3	<i>Polaris</i>	316
7.3.4	<i>Apocalypse World</i>	319
7.4	Bringing in the Digital	325
7.4.1	<i>Elegy for a Dead World</i>	326
7.4.2	The Squinkifer Oeuvre	326
7.4.3	<i>Sleep is Death</i>	328
7.5	What does it mean to collaborate?	330
7.5.1	Three Questions	330
7.5.1.1	Why would we want to collaborate with a system?	330
7.5.1.2	Is “collaboration” a mismatched metaphor?	332
7.5.1.3	What kinds of human/digital collaboration are possible?	336
7.5.2	Naïve Collaboration	336
7.5.3	Deep Collaboration	340
7.6	Futures for Digital Collaborative Storygames	343
7.6.1	Systematizing Naïve Collaboration	343
7.6.2	Exploring common ground	345
7.6.3	Deeper Generation	347
8	Collaborative Storygames Case Studies	351
8.1	<i>Problem Planets</i>	351
8.2	<i>SPUN Adventure</i>	353
8.3	Minimalist Story Generator #1	355
8.4	<i>Ice-Bound</i> as Collaborative Storygame	357
8.5	<i>Almost Goodbye</i>	361
8.6	<i>The Prisoner In Block Nineteen</i>	363
8.7	<i>Archives of the Sky</i>	364

9 Conclusion	367
Appendix A Project Yarn	374
Appendix B Archives of the Sky	388
Ludography	404
Bibliography	411

List of Figures

0.1	A performance of the procedural play <i>Bad News</i>	7
0.2	A game of <i>Downfall</i> in progress.	8
1.1	<i>Adventure</i> , the original storygame.	19
1.2	The character creator in <i>Mass Effect: Andromeda</i> , an example of expressive input.	25
1.3	The game genres, according to games distribution service Steam.	30
1.4	Choice graph from <i>The Cave of Time</i> , Choose Your Own Adventure #1.	39
2.1	Various adventure games.	45
2.2	A puzzle integrated into the story world in <i>Riven</i>	50
2.3	Scenes from <i>King's Quest IV</i>	58
2.4	The Adventure Triangle	60
2.5	Edges on the adventure triangle.	62
2.6	Promises and challenges on the adventure triangle.	63
2.7	Screenshots from <i>The Witness</i> , which sacrifices Story to focus on Puzzles and Exploration.	71

2.8	“Story dump” audio logs from <i>The Witness</i> and <i>Singularity</i>	75
2.9	Screenshots from <i>Firewatch</i> , which sacrifices Puzzles to focus on Story and Exploration.	80
2.10	Screenshots from <i>Her Story</i> , which sacrifices Exploration to focus on Puzzles and Story.	87
2.11	Three different styles of game sharing the same eureka story.	96
3.1	Screenshot of <i>Winterstrike</i>	100
3.2	Raymond Queneau’s “Cent mille milliards de poèmes” (Hundred Thousand Billion Poems), (1961)	108
3.3	Screenshot from <i>Save the Date</i>	116
3.4	Screenshot from <i>Fallen London</i>	119
3.5	Screenshot from <i>King of Dragon Pass</i>	122
3.6	Screenshot from the unreleased <i>Storyteller</i>	123
3.7	Screenshot from the author’s minimalist sculptural fiction game <i>Perfect</i>	125
3.9	ChoiceScript code controlling a sentence describing the first day of shooting in <i>Hollywood Visionary</i>	132
4.2	Themes and tags used in <i>The Ice-Bound Concordance</i>	155
4.3	<i>Ice-Bound</i> sockets.	157
4.4	<i>Ice-Bound</i> symbols triggering events and endings.	159
4.5	Complex ending lexia from <i>The Ice-Bound Concordance</i>	161
4.6	Example rendering of complex <i>Ice-Bound</i> ending lexia.	162

4.7	Resolving endings with the printed book in <i>The Ice-Bound Concordance</i> .	167
4.8	Screenshot from final level of <i>The Ice-Bound Concordance</i> .	170
4.9	Visualization tools for <i>Ice-Bound</i> .	175
5.2	Screenshot of <i>Prom Week</i> .	192
5.3	Example <i>Prom Week</i> instantiation.	193
5.4	Screenshot from <i>Blood & Laurels</i>	200
5.5	Example Praxis syntax.	202
5.6	Example Prompter syntax.	202
5.7	Detail of <i>Blood & Laurels</i> interface.	204
5.8	Screenshot from <i>Redshirt</i> .	208
5.9	<i>Redshirt</i> uniquely allows for the generation of a new social world with specific overall qualities.	210
5.10	Screenshot from <i>Parable of the Polygons</i> .	235
5.11	Social simulation play time as a function of complexity of the simulation.	237
6.1	Text messages communicating the social state in <i>Prom Week</i> .	274
6.2	A sample schema for a role-playing domain in Ensemble.	282
6.3	The Ensemble Tool: Rules Editor view.	283
7.1	Back of the box for <i>Tales of the Crystals</i> .	287
7.2	A selection of cards from <i>Once Upon A Time</i> .	294
7.3	A modern Wunderkammer.	297

7.4	Collaborative storygames at the center of Simulative and Performative spectra.	300
7.5	A game of <i>Microscope</i> in progress.	312
7.6	The GM moves in <i>Apocalypse World</i>	323
7.7	<i>Coffee: A Misunderstanding</i> in progress.	327
7.8	Screenshot from <i>Sleep is Death</i> , showing the “wizard” player’s interface for modifying the scene.	329
7.9	A page from <i>Annals of the Parrigues</i>	334
7.10	A table showing Generation of ideas from 1st edition <i>Dungeons & Dragons</i> .337	
8.1	Example <i>Problem Planets</i> Story.	354
8.2	Hypothetical transcript from <i>SPUN Adventure</i>	356
8.3	Basic system architecture for <i>Almost Goodbye</i>	362
9.1	From <i>The Knight of the Burning Pestle</i>	369
A.1	Screenshots from Project Yarn.	378
A.2	Results of an action, showing social state causing gameplay consequences. 379	
A.3	A sample social graph in Project Yarn.	381

List of Tables

1.1	Examples of player interaction at different level of abstraction.	33
4.1	Some of the template commands used in <i>The Ice-Bound Concordance</i> . .	164
4.2	Example <i>Ice-Bound</i> template, showing pronoun replacement.	165
6.1	Statistics for <i>Prom Week</i>	265

Abstract

Changeful Tales:
Design-Driven Approaches Toward
More Expressive Storygames

by

Aaron A. Reed

Stories in released games are still based largely on static and predetermined structures, despite decades of academic work to make them more dynamic. Making game narratives more playable is an important step in the evolution of games and playable media as culturally relevant art forms. In the same way interactive systems help students learn about complicated subjects like physics in a more intuitive and immediate way than static texts, more dynamic interactive stories open up new ways of understanding people and situations. Such dreams remain mostly unrealized in released and playable games.

In this dissertation I will describe a number of design and technical solutions to the problem of creating more expressive and dynamic storygames, informed by a practice-based approach to game production. I will first define a framework for the analysis of games, including especially the terms *storygame* (a playable system with units of narrative where the understanding of the interconnectedness between story and system is crucial) and the notion of *narrative logics* (the set of processes that define how player input affects the next unit of story presented by the system). I will exercise

this framework on an existing and well-known storygame genre, the adventure game, and use it to make a number of claims about the mechanics and dynamics of narratives in this genre that are borne out by an analysis of how contemporary games adopting some of its aesthetics succeed and fail. I will then describe three emerging storygame modes that are still in the process of being defined, developing a critical framework for each informed by close readings and historical analysis, and considering what design and technical innovations are required to fully realize the new mode's potential. These three modes I discuss are sculptural fiction (which shifts the focus from navigating to building a structure of narrative nodes), social simulation (games that explore the possibility space created by a set of simulated characters and rules for social interaction), and collaborative storygames (in which the lexia are generated at least in part by the participants during play). Each theoretical chapter is paired with a case study of one or several fully completed and released games I have created or co-created in that mode, to see how these design ideas were realized and technical advancements implemented in practice. I will conclude each section with applied advice for game makers hoping to work in these new spaces, and new technological developments that will help storygames continue to evolve and prosper.

Acknowledgments

This dissertation would not have been possible without my tremendous collaborators and lab mates at UC Santa Cruz, including Matthew Balousek, Bryan Blackford, Duncan Bowsman, Kate Compton, Melanie Dickinson, Teale Fristoe, Elaine Gan, Chaim Gingold, John Grey, April Grow, Sarah Harmon, Eric Kaltman, Kathleen Kralowec, Larry LeBron, Chris Lewis, Heather Logas, Paul Maddaloni, Marcelo Viana Neto, Stacey Mason, Peter Mawhorter, Josh McCoy, Wes Modes, John Murray, Joe Osborn, Allie Riggs, James Ryan, Serdar Sali, Ben Samuel, Adam Smith, Gillian Smith, Ben Spalding, Dietrich Squinkifer, Anne Sullivan, Brandon Tearse, Michael Thomét, Phoenix Toews, Mike Treanor, and Dani Williamson. It is not an exaggeration to say each of you taught me something (and likely many somethings) that helped make this project what it is.

I've also had the extraordinary luck to be mentored (or at least gently encouraged) by many generous folks in academia or industry, including Sandy Baldwin, Kathi Berens, Dan Fabulich, Clara Fernández-Vara, Dene Grigar, D. Fox Harrell, Ian Horswill, Arnav Jhala, Mark Marino, Nick Montfort, Richard Rouse, and Jichen Zhu. Thanks for your encouragement, your favors, for listening, and for caring. Especial thanks to Michael Chemers for serving on my committee, being the voice of KRIS, and teaching me about monsters and robots.

Emily Short served on my advancement committee, and would most likely have been on my dissertation committee had she not already offered me a job by that point. Thanks for all the encouragement, feedback, and friendship over the years.

My advisors Noah Wardrip-Fruin and Michael Mateas suffered with me through two straight graduate degrees, helping me learn to navigate the treacherous waters of being an artist and an engineer, sequentially at times but more often simultaneously. Seven years was a long enough apprenticeship in the Middle Ages to become a master craftsman, so perhaps this shouldn't be a surprise, but: I have learned more from you both about this field of study and many subjects besides than I could ever have expected. It has been an honor to learn, teach, and play with you.

My mom and dad have always been ready with a sympathetic ear or a pep rally when I needed a boost, which it turns out is often when you're working on a PhD, and this dissertation is dedicated to them.

Finally, to Jacob, my collaborator, friend, and partner: I couldn't have done it without you.

Note: this dissertation uses the singular *they* throughout whenever a gender-neutral pronoun is required. If the reader feels compelled to point out this goes against the APA style otherwise followed here, they are welcome to mentally swap in the gender-neutral pronoun of their choice instead.

Introduction

I doubt not but some man of severe judgment so soon as he hath once read the title of this book will immediately say that I had more need to exhort men to work, than to teach them to play; which censure, if it proceed not of such a forward morosity that can be content with nothing but that he doth himself, I do not only well admit, but also willingly submit myself thereto. And if I could be persuaded that men at my exhortation would be more diligent to labor, I would not only write a treatise twice as long as this, but also think my whole time well bestowed if I did nothing else but invent, speak, and write that which might exhort, move and persuade them to the furtherance of the same. But if after honest labor and travail recreation be requisite (and that need no further probation because we favor the cause well enough), I had rather teach men so to play, as both honesty may be reserved, their wits exercised, they themselves refreshed, and some profit also attained, than for lack of exercise to see them either pass the time in idleness, or else to have pleasure in things fruitless and uncomely.

From The Most Noble, Auncient, and Learned Playe, Ralph Lever and William Fulke, 1563.

In April 2017, as I was nearing a final draft of this dissertation, one of the most widely read figures in games academia wrote a provocative¹ essay for *The Atlantic* called “Video Games Are Better Without Stories.” Calling into question games’ obsession with narrative, the piece makes a case that the medium is not inherently well suited for storytelling, revisiting an old debate.² Ian Bogost argues that every medium used to tell

¹Among the many counterpoints that followed in the wake of this piece, one of the least angry is Walker (2017).

²A quick recap of which can be found in (Juul, 2005, p. 15-17).

stories is built from particular technical and aesthetic foundations. He posits that games are fundamentally about the simulation of spaces and objects within them, and that this is a fundamentally poor platform for narratives. Bogost sees the “environmental storytelling” of 3-D games as limited in sophistication (comparing the best of them to young adult novels) and worse, inherently flawed: “Are the resulting interactive stories really interactive, when all the player does is assemble something from parts?” Praising a new game in this tradition, *What Remains of Edith Finch*, Bogost points out that its innovations are enabled through an exploration of the aesthetics of its engine, and its principal accomplishments are not narrative:

Rather, they are novel expressions of the capacities of a real-time 3-D engine. The ability to render light and shadow, to model structure and turn it into obstacle, to trick the eye into believing a flat surface is a bookshelf or a cavern, and to allow the player to maneuver a camera through that environment, pretending that [it’s] a character. *Edith Finch* is a story about a family, sure, but first it’s a device made of the conventions of 3-D gaming.

Bogost’s provocation echoes another existential crisis for gamemakers from a decade earlier, though from a different kind of figure. Roger Ebert, undoubtedly the best-known living film critic at the time, came out in 2005 against the growing sentiment that games might too be an artistic medium deserving of deep thinking and critique (as a casual aside in a column answering letters from readers). Ebert, as did some games scholars at the time, identified player agency as fundamentally incompatible with good stories:

Video games by their nature require player choices, which is the opposite of the strategy of serious film and literature, which requires authorial control. I am prepared to believe that video games can be elegant, subtle, sophisticated, challenging and visually wonderful. But I believe the nature of the medium

prevents it from moving beyond craftsmanship to the stature of art. To my knowledge, no one in or out of the field has ever been able to cite a game worthy of comparison with the great dramatists, poets, filmmakers, novelists and composers. That a game can aspire to artistic importance as a visual experience, I accept. But for most gamers, video games represent a loss of those precious hours we have available to make ourselves more cultured, civilized and empathetic.

Ebert claimed art in games is compromised because player agency will trump artistic intent. Bogost says much the same thing to discredit story in games, dismissing the interactive narrative *Façade*, for instance, because it allows the player agency to act in a manner that breaks narrative immersion. Like Ebert, Bogost feels game stories are less mature than those told in other media: his claim that “the best interactive stories are still worse than even middling books and films” mirrors Ebert’s “no one [can] cite a game worthy of comparison with the great dramatists, poets, filmmakers...” Games in general (according to Ebert) and story in games (according to Bogost) are fundamentally flawed by their inherent structure, and besides have yet to demonstrate that they can live up to the narrative quality of other media.

Bogost and Ebert, of course, come from very different places. Ebert was the ultimate outsider to games; Bogost has been playing, making, and thinking about them for two decades, has been highly influential on how academics think about the field, and has even helped found genres of gaming, such as news games and serious games. Whereas Ebert’s argument was easy to dismiss for being ill-informed,³ Bogost’s might give us more pause. In addition, Bogost’s article comes twelve years later. In response

³Ebert famously admitted in a follow-up piece five years later (2010a) that his opinions on video games came from watching talks about them or seeing screenshots: he had never actually *played* any of them—something which, to his credit, he admitted was a mistake in a follow-up column (2010b).

to Ebert then and again to Bogost now, many gamemakers have compared the medium to the early days of cinema: we're still learning what works and what doesn't, and the games of the future will doubtless have more mature, sophisticated, and dynamic stories. With twelve more years for games to continue maturing, is this argument growing hoary?

While the argument for quality is subjective, Bogost's claim about the foundations of games has a basic flaw. Speaking of other media (poetry, painting, film) he says we can think of these as "the aesthetic form of common materials" of which they are made, and upon which structures like narrative can be formed: "the stories come later, built atop the medium's foundations." He then makes a case that the foundation of gaming as a medium is a simulation of space and objects: games are most successful at "taking the tidy, ordinary world apart and putting it back together again in surprising, ghastly new ways." Bogost's argument fails to consider games that are *not* about simulations of spaces. But more seriously, he fails to bring his central point full circle and apply it back to games. Having gained a deep understanding of how games work, many gamemakers are in fact using them to tell stories in ways unique to (and driven by the strengths of) the medium. "The stories come later, built atop the medium's foundations." Bogost's conclusion that games should "abandon the dream of becoming narrative media" seems as suspect as a 1980s TV critic arguing that television as a medium would never surpass the sitcom: it's both looking in the wrong place for innovation and failing to account for the inevitable technical and artistic sea change that comes with a new generation's deeper understanding of a medium and its foundations, technical and artistic. The fact that this conclusion can still be argued for, however,

speaks to how much work is left to do in developing storygame theory and bringing more effective storygames to release and mainstream discussion.

Bogost is right that most big games do not approach story from a deep understanding of ludic and narrative theories and how they can intertwine. But much of the design innovation and experimentation that drives this knowledge forward is happening outside the mainstream games industry, outside the academy, even outside the space of digital games entirely. Much of it is happening in games that are not built around simulations of spaces or objects, and some of it troubles the definition of “story” we might comfortably apply to narratives in other media, specifically because it is often built on game-like—on playful—foundations.

Consider *18 Cadence* (discussed further in 3.3.2), a project I released in 2013 about exploring the history of a house. The player can move through the rooms of the house, but can also move through two additional dimensions: time, and perspective. They might see what was happening in the house’s kitchen in 1944, through the eyes of either a war-weary father arguing with his youngest son not to enlist, or the sixteen-year old shaking with rage that he’s not fighting with his brothers. They might examine the continually changing objects showcased in the living room, from a fireplace to a large wooden radio to a black-and-white television to a Nintendo game console. These fragments of story, setting, and character can be collected on a central workbench, where they can be reordered, reassembled, and recombined. In *Cadence* there is not a preassembled story graph to navigate, and indeed the hundreds of textual fragments do not inherently create a meaningful story in aggregate. It’s up to the player, in the role

of historian, curator, and editor, to assemble a story from this raw material: to discover in the history of the house and all the people who’ve lived there some juxtaposition or sequence of events, objects, and personalities with personal meaning.

Cadence is an example of *sculptural fiction*, one of three emerging modes of interactive story I will discuss in this dissertation. Sculptural fiction turns the primary mechanic of play in a storygame from navigating a fixed structure to assembling one out of narrative pieces. It answers Bogost’s challenge about whether piecing together story is truly interactive by finding meaning in the act of assembly, and by giving players tools to build personal and expressive conclusions from fragments of story curated by an author.

Or: consider that *Tactics* (1954), a boardgame that launched the hobby of wargaming (which would eventually evolve into things like *Dungeons & Dragons* and both the analog and digital roleplaying game genres) was designed by its creator as a way to teach himself about combat. Facing the prospect of being summoned for a tour of duty in the Korean War, Charles Roberts built a game to guide his thinking about “the mechanics of war” (Peterson, 2012, p. 2). This impulse—to understand through simulation—has become easier to do as personal computers have put even complex simulations at our fingertips. Modern games of *social simulation*, the second mode I’ll discuss, aim to explore characters through positing, encoding, and interacting with simulations of the mechanics of social interaction. As tools for expression, experimentation, and story building, social simulations provide a different kind of simulative framework for games than one focused on a 3-D environment. Contemporary games such as *Bad News*, an experimental play



Figure 0.1: A player (right middle) and performer (right top) during a performance of the procedural play *Bad News*, driven by the “Talk of the Town” social simulation engine; system creator James Ryan (left) drives the simulation and feeds character notes to the performer.

from my colleagues at UC Santa Cruz driven by social simulation, or the miniature social simulations from designer Nicky Case (discussed in 5.5.2), or the emergent social strategy involved in a playthrough of *Blood & Laurels* (5.3.2), demonstrate how making playable simulations of character can change the way we think about people and fictions.

The tabletop roleplaying movement descended from *Tactics* has gone through a number of revolutions, but in the last decade a particularly fascinating transition has happened. An indie roleplaying movement has drifted away from the aspects of simulation that computers are good at, and moved from pre-planned scenarios toward games focusing on stories that are created together by all participants. Caroline Hobbs’ *Downfall* (2015), for example, asks three players to imagine a flourishing society based



Figure 0.2: Index cards tracking characters, traditions, and the Haven during a game of *Downfall*. Image courtesy contessa.ocks.

around a fundamental flaw (such as Greed, Nationalism, or Ambition) and then tell, through a series of scenes, how that society collapses. The rules and mechanics are carefully structured to enable even players untrained in storytelling to develop and enact such a narrative. For instance, the players create three characters with different perspectives on the society: one who comes to understand the flaw and works to stop it, one who actively promotes its growth for their own gain, and another who just wants to maintain the status quo and keep things from changing. Other mechanics help advance the story and move it towards a narrative of collapse.

Downfall is an example of a *collaborative storygame*, the third emerging mode

I'll discuss. While most such games to date have been analog, these games have made fascinating design innovations in the space of how people can tell stories together, techniques which digital games designers would be remiss to overlook. Indeed, bringing in these insights from adjacent corners of game, play, and story theory will frequently point the way towards how these elements can be more effectively intertwined, opening us up to new visions of how games might be than we might find by limiting our focus to mainstream digital games.

In this dissertation I will explore the three emerging modes of interactive narratives mentioned above: sculptural fiction, social simulation, and collaborative storygames. Each is a new way of thinking about and telling stories in games, and none has yet been comprehensively analyzed to find out how it works. A focus on completed, released, and playable games is intentional: learning from Ebert's mistake, our discussion will be based whenever possible on games that have been played, rather than academic systems not integrated into released games or purely theoretical work not yet field tested. I am both a theorist and a designer, and also a technologist: across this dissertation I will discuss over a dozen games I have designed and built (alone or in collaboration) that connect to my theoretical framework, most of which have also been released, reviewed, and played in the wild.

This focus not only grounds discussion but helps point toward the kind of insights likely to have actual impact on game design. Creating and exhibiting experimental games provides a "proof by existence" to the wider games world that ideas are workable and can drive a successful game. Many groundbreaking games, from *SimCity* to *Flower*

to *Dear Esther*, were released in part as statements about what games can be (sandboxes; relaxing; free of obstacles). Sometimes such games go on to spawn whole new genres as gamemakers see both a new possibility space and a working demonstration that a successful game can be made in that space. This demonstrative work is important to pushing the discourse around playable media forward, and is an essential outward-looking companion to more inward-focused theoretical work.

In chapter 1, I will begin by defining a basic framework for our discussion of storygames, including a definition of the term and a set of related terms that will be useful throughout. In particular, I will define the term *expressiveness*, referring to a player's ability to make a creative contribution, and *narrative logics*, the set of operations that drive how the pieces of story in a game are presented to and manipulable by the player.

Chapter 2 will test drive this framework by using it to do a deep analysis of one of the most famous (and famously failed) existing forms of storygame, the adventure game. By using the established framework, I will construct a bottom-up definition for the genre that focuses on how the story is played, including a historical survey of definitions of the form leading to a broader working definition than is generally accepted: I will show that many kinds of both graphical and text-based games, while often previously studied as separate phenomenon, have no real design differences in their narrative mechanics and logics. This bottom-up approach will be used throughout the dissertation to group games based on how their narrative is playable, rather than what surface or implementation features they have in common. Using my framework and definition, I will then posit

that adventure games have a fatal flaw which in part led to their extinction: a trio of dynamics (exploration, puzzles, and story) which are connected such that having narrative logics that effectively join all three is extremely difficult. I will then show, however, how a deep understanding of adventure games can guide our interpretation of contemporary games and point towards new design directions, through a close reading of three recent games (*Firewatch*, *The Witness*, and *Her Story*). Each of these removes a different foundational component of adventure games to focus on the logics that amplify the connection between the remaining two.

For the rest of the dissertation, I will alternate a design chapter fleshing out one emerging form of storygame with a case study chapter describing a major project or projects I have created to explore it. I will consider foundational experiments establishing each form's boundaries, narrative mechanics and dynamics that work well with it, and challenges for authoring. The case study chapters will examine how these games operate at a detailed level in order to exercise each platform as a framework for storygame construction. I will also detail new technologies developed to realize these novel design ideas, as part of my technical contribution to interactive story research. These specifics will allow me to provide practical and field-tested examples for the ideas outlined in each preceding chapter.

Chapter 3 begins with sculptural fiction, where the player primarily builds a story structure out of narrative pieces, rather than navigating a pre-built structure. I will trace the pre-digital origins of sculptural fiction through pattern poetry, fragment novels, and into the computer age with hypertext and modern implementations of nonlinear

systems based on logics other than branching paths. I will then discuss the challenges with authoring sculptural fiction and new ways of thinking to help overcome them. Chapter 4 follows this with a case study of *The Ice-Bound Concordance*, a full-scale game constructed with sculptural fiction ideas in mind.

Chapter 5 will discuss social simulations, games that explore the possibility space created by a set of simulated characters and rules for social interaction. I will discuss how content authored for social simulations must work for patterns of social interaction rather than specific characters, which connects the mode to literary traditions such as the comedy of manners in which the focus is less on specific characters and more about how they fit into (or oppose) existing social structures. I will discuss three major released games based around social simulation (*Prom Week*, *Blood & Laurels*, and *Redshirt*) and take lessons from all three to look at the practical considerations of authoring for this kind of storygame. In chapter 6 I will look more closely at the authoring process for *Prom Week*, including how the lessons learned from its creation gave rise to a new tool for crafting social simulations, Ensemble.

In the final pair of chapters, I will discuss collaborative storygames, in which the lexia are generated at least in part by the participants during play, guided by systems (analog or digital) designed to produce a compelling story. Chapter 7 studies this genre's mostly analog history in detail, placing it within existing frameworks such as tabletop roleplaying theory and applying our storygame framework to it as well. I will discuss four analog collaborative storygames in depth: an untitled journal-writing game by Andrew Plotkin, the games *Microscope* and *Polaris*, and the *Powered by the Apocalypse*

roleplaying engine. I will look at several digital experiments in collaborative storygames, and discuss in depth what lessons digital games can learn from this genre, in the hopes of pointing the way towards genuinely new kinds of mechanics and logics that enable collaboration between humans, and between humans and systems. Finally, in chapter 8 I will consider a number of small projects I have made exploring notions of collaboration, including reconsidering *The Ice-Bound Concordance* as a prototypical collaborative storygame.

Across these projects, my goal is in part to develop a changeful aesthetic for interactive stories. Bogost is correct that many game stories are not meaningfully interactive: they strafe against the limitations of a playable medium, rather than working with them. They would, in fact, best be told in another medium. Noah Wardrip-Fruin has written about “textual instruments” (discussed further in 7.5.1.1) which can be *played* in a deeper sense of the word, and upon which a performer can create something genuinely original but only made possible by the instrument’s designer. The practical problem of how to create such instruments has rarely been formally explored by game designers. Theater historian Michael Chemers has described the need for dramaturgs to understand the uniqueness of their art form: its theatricality, which “specifically takes advantage of those qualities of the theater that no other medium can reproduce” (2010, p. 88). He finds that “Unquestionably, the most theatrical thing is change.” Games are not theater, but they have in common the coming-together of multiple art forms, and the “creation of a work of living art” unique to each play session. Chemers writes

that “a play is a machine that manufactures meaning” (p. 69) through the complex collaboration of (among others) the playwright, the actors, the director and dramaturg, and the audience. A changeful storygame likewise brings systems, authors, and audience together to manufacture meaning. As we continue to innovate in both game design and game technology, we have the potential to make this collaboration even more close, and increasingly, more personally meaningful.

My title for this work, *Changeful Tales*, points toward a new way of thinking about stories in games as fundamentally pliant, reflective, meaningful and playful: as instruments with which we can perform genuinely interactive stories, and as machines for making highly personal meanings. A flute enables changeful play; so too do the rules of *Downfall*, enabling its players to perform ephemeral stories of doomed civilizations. But so also do the engine and fiction together of *Bad News*, asking its players to find meaning in a network of generated relations between randomized characters; and, I hope, my own *18 Cadence* (discussed in 3.3.2) in asking players to perform meaning from the raw materials of a hundred years of mostly trivial events.

These ideas are not “new” in and of themselves: even some of the earliest storygames have taken part in exploring and developing this aesthetic. Where I hope to contribute is by embracing and deeply considering how changefulness can be a new kind of foundation for thinking about storygames, informed by deep technical understanding, practice-based wisdom from actual game creation, and rich design thinking based on decades of game design theory—by, as Bogost asks us to, building atop the medium’s foundations.

Chapter 1

Framework

"I KNOW OF PLACES, ACTIONS, AND THINGS."

From *Adventure* (1977), in-game instructions

1.1 Vocabulary

Game studies, while gaining its footing, is still a young field, and many terms of art are still in the process of establishing fixed definitions. In this chapter I will both assemble a vocabulary of terms used in this document (some existing, and some new) and provide some theoretical groundwork for the analysis and discussion to follow.

1.1.1 Foundational terms

I will use the term **player** to describe the person interacting with a playable system. Most of the systems I'll discuss will be games or **playable media** (Wardrip-Fruin, 2005), a broader term used less to categorize than as a lens to study the way an interactive system might be played with. Not all playable media include game-like

features such as conflict, goals, or quantifiable outcomes, but we explore both games and playable media through playful interaction, as players in a broad sense of the term.¹ Specifically, my focus is playable systems with significant narrative elements. The adjective **ludic** refers to a system’s playful or game-like qualities, as opposed to its narrative or story-like ones. Aarseth’s term **ergodic literature** (1997), for a text that requires nontrivial (often ludic) effort from the reader in order to keep advancing, helps us distinguish these kinds of works from their more traditional static ancestors. We might also make use of his distinction (p. 6; borrowed from Doob 1990) between unicursal works that can be read in only one order, and **multicursal** works through which more than one path can be taken.

Players, unlike readers or viewers, can encounter sections of a multicursal work in different orders from each other, or see a different set of sections entirely, based on their ludic interactions. One player’s experience may vary significantly from another’s, or the same player may have different experiences across two different sessions. Montfort’s term **traversal** describes a specific path taken through a multicursal work, from a beginning to an ending. Montfort also defines a successful traversal as one corresponding to a

¹I’m going to skip a discussion of the definition of *game* and *player* here, both since I’ve done it before (2011b) and because many others have done a more thorough job (especially useful is Juul 2005, p. 23-47). *Play*, of course, is also a complex term. Another useful lens for us is Chaim Gingold’s work on “play studies” (2016), centering play experiences over game artifacts:

Rather than see these activities and objects as liminal or degenerate games (as Juul’s scheme does), or allow the definition of “game” to slip and apply to playful activities which are not actually games (as is common usage), I wish to consider them on their own terms, as play. (p. 2)

Gingold’s focus transcends digital media to include playground games and other non-artifactual practices. He acknowledges the difficulties caused by the lack of an English word for “things which can be played,” relying on terms such as *toys*, *playthings*, and *play experiences*. While most of what I’ll discuss here broadly qualifies as “game,” Gingold’s framework is a reminder of the more expansive scope I mean to include.

win condition. I would like to instead use the broader term **satisfying traversal** for the situation where both the player and author are in general agreement that the work was encountered as intended, and need not necessarily be replayed. In an unsatisfying traversal, the experience was cut short or otherwise ended in a frustrating manner for either player or author.²

I will use the term **player character** to refer to a game character controlled by a player, and the more specific term **avatar** when this character is a visible entity in the game world. **Non-player characters (NPCs)** are everyone else in the story.

The term **lexia** is used in hypertext theory to signify a block of text of arbitrary size (Landow, 1994). In a traversal of an interactive story, pre-authored lexia are often experienced by the player in a dynamic order. Even if the content in question is not strictly or entirely text, I'll use the term here to mean any atomic chunk of content a storytelling system can make use of. Aarseth (1997, p. 62) distinguishes between the static pieces of text within a system and the potentially more dynamic text that might be encountered at runtime, possibly assembled from pieces of static texts or generated through some other process. For static texts I prefer to simply say “lexia”; the dynamic ones, which Aarseth calls *scriptons*, I find it useful to divide into two categories: **templated lexia**, which are pieces of static text with slots that can be dynamically filled, and **generated output**, for any other kind of text assembled or reconfigured in some more complex fashion.

²For an overview of research into whether and when players feel satisfied with their reading of an interactive story, and the circumstances in which replay is required for readers of such stories to feel closure, see Mitchell and McGee (2012).

Finally, when we speak of the fictional world within a narrative game, the term **mimesis** refers to the illusion that that world is a consistent, existing place. Something that breaks mimesis reminds us we are just playing a game, which can interfere with our immersion in the story (but can also be used for artistic effect).

1.1.2 Storygames

This dissertation primarily concerns interactive stories, for which my preferred term is **storygames**. Originating in one of the first academic works about textual computer games (Buckles, 1985), the term has since been used in various ways and contexts.³ My usage is meant to succinctly suggest an artifact with both narrative and ludic elements that are closely intertwined. Specifically, I define a storygame as:

- a playable system, with
- units of narrative, where
- the understanding of both, and the relationship between them, is required for a satisfying traversal.

³Buckles’ original definition is simply that of a story embedded in a game, which with the benefit of hindsight seems too general: although in practice her use of the term suggests that “rather than one element being embedded in the other, both are essential to the experience and are intertwined rather than nested” (Montfort, 2004).

Murray (2004) writes about the awkwardness of the term (or its cousin “game-story”) as indicative of an evolving technology for which a unique vocabulary has not yet emerged, citing the early use of “photo-play” in film history as another example. She offers the term “cyberdrama” but this is itself an awkward portmanteau, now dated somewhat by the aging *cyber-* prefix.

Outside of academia, “storygame” has been used inconsistently: to refer specifically to games which feature both a simulated world and a guarantee of generating a compelling story (Maher, 2008), or to tabletop roleplaying games that put more of a stress on narrative than mechanics (Ashwell, 2016) sometimes called Narrativist games (Edwards, 2001a) and which I define as a subset of storygame (see Chapter 7); to refer to specific categories of interactive story systems such as choice-based narratives or even as a generic term for any game with narrative content.

While the term is certainly confused and overloaded, it has the virtue of being short, catchy, and immediately understandable to laypeople, so storygame it is. My own first use of the term in the sense described here was in (2014a).



Figure 1.1: *Adventure*, the original storygame. Image courtesy User:Autopilot / Wikimedia Commons / CC-BY-SA-3.0.

1.1.2.1 *Adventure* as prototypical storygame

Consider the original parser-driven interactive fiction game *Adventure* (1977), which inspired Buckles' term. It is clearly a playable system: the user can input commands that change the state of an underlying simulation. It also clearly contains elements both of narrative (the discovery and exploration of a magical underground cavern; dragons and dwarves) and gameplay (points scored by returning treasures to a certain location; a light source necessary for movement that expires after a certain number of turns).

How are the understanding of this story, its gameplay, and their interconnectedness required for a satisfying traversal of *Adventure*? We might say that understanding the narrative elements of the game helps the player's ludic performance within the system, and understanding the ludic elements helps make sense of the narrative. Together, this two-way dependency enables a satisfying traversal.

For example, a player of *Adventure* might encounter some narrative text describing a thirsty plant. This suggests that certain ludic actions might advance the story, such as searching for an empty bottle, filling it with water, bringing the bottle to the vicinity of the plant then watering it.⁴ The descriptions of the items in question are also pieces of narrative content which help players discover the above ludic sequence. Had the description of the plant or any other item been replaced by nonsense text or arbitrary labels, the correct sequence of commands would be difficult or impossible to discover.⁵

As a player comes to understand the ludic functioning of the game, their understanding of the narrative deepens in turn. While the introduction text to *Adventure* claims that `MAGIC IS SAID TO WORK WITHIN THE CAVE`, it is only through finding magical words inscribed at specific locations, typing them in, and observing how they affect the simulation that the player gains a true understanding and appreciation of how

⁴Though described narratively here, the player is of course actually inputting a set of specific ludic commands such as `GO EAST`, `GET BOTTLE`, and `WATER PLANT` to realize this plan.

⁵Does this mean a game like the interactive fiction *The Gostak* (2001), where the player is presented with an environment described in an unfamiliar fictional language and must learn through trial and error how to proceed, is not a storygame? No: it just means the barrier to understanding the connection between the unfamiliar narrative elements and the ludic systems is higher. In fact, *The Gostak* is carefully designed to allow the player to discover the meaning of its nonsense nouns and verbs through experimentation.

magic functions in the story. Coming to understand the ludic mechanism of your lamp's finite battery life adds a narrative element of tension, while the discovery of the vending machine in the maze introduces the possibility of both mechanical and narrative reprieve. The code managing the various dwarves stalking and attacking the player makes the threat implied by their descriptive text have actual consequences within the simulation, though the existence of a mechanic for resurrection gives *Adventure* a more playful tone to counterbalance what might otherwise be a grim and difficult storygame. And on a simpler level, the mechanics for movement through mappable space, simulations of light and dark, doors that can be unlocked and food and water that can be consumed all help a story of exploration in a magical cave become actualized, above and beyond the literal string contents of the messages the game displays. The ludic mechanisms and narrative content, and the way they interrelate, are both required for a successful traversal.

1.1.2.2 What Isn't a Storygame?

My aim is to use the term “storygame” as a lens rather than a set of qualifying rules: whether something meets its exact requirements is less interesting than whether we can gain useful insights by peering through the lens. However, identifying kinds of games for which the lens seems less useful is a worthwhile exercise of the way I've defined the term.

Clearly, a game with no narrative component at all, such as the abstract strategy game *Threes*, is not especially useful to discuss as a storygame. Nor are artifacts with no

ludic components, such as traditional novels.⁶ We can also exclude games where story and gameplay are both present but not connected. This is true of the game *SpaceChem* (2011) where story lexia appear in blocks of text between levels: these lexia can be ignored entirely without impacting the chances of a successful traversal, and nothing about the gameplay especially affects the player's perception or understanding of the story.

What about a game like *Super Mario Bros.* (1985), which presents story content in the form of recognizable items such as mushrooms, turtles, castles, and princesses, as well as occasional in-game messages implying a narrative, such as **THANK YOU MARIO! BUT OUR PRINCESS IS IN ANOTHER CASTLE!** While this narrative content might contextualize, motivate, or comment on the gameplay, I would argue that actually *understanding* it is not required for a satisfying traversal.⁷ This is in part because *Mario*'s small range of possible inputs and simulated entities means a player can quickly discover the ludic consequences of all possible interactions, and learn even without any narrative assistance what is required for a successful traversal.⁸ A version of *Mario* with all graphical elements replaced by random images might be more initially disorienting,

⁶Certainly readers of novels engage in practices such as interpretation that can sometimes feel game-like, as in murder mysteries; but (Aarseth, 1997) and (Eskelinen, 2001) make strong cases that interpretation is a different sort of act from one capable of altering specifics of the text being encountered.

⁷The same argument can be applied to metaphorical readings of abstract strategy games that attempt to explain their ludic actions within a narrative context, such as a reading of *Tetris* (1986) as symbolic of harried modern American life (Murray, 1997, p. 144), or as a game about stacking bodies in a mass grave (Koster, 2013, p. 172). In both cases, knowledge (or lack thereof) of the narrative interpretation does not affect the player's ability to traverse the work; and while a player may choose to alter their play style in response to an interpretation, the game will not respond to this change in strategy.

⁸For instance, even a player who did not know what a mushroom was would appreciate how colliding with one increases Mario's power; and though a goomba was not a recognizable object before the game's release, through observing its effects when collided with from the side (the avatar's death) and above (the goomba's destruction) players could understand that its ludic role was that of an enemy to be defeated.

but still perfectly playable after a few minutes of experimentation. Similarly, the game of chess is given a patina of story through the names of its pieces, and players might draw parallels between certain mechanics and certain piece names: but chess is also perfectly traversable without understanding the significance of those names or their historical context (Treanor and Mateas, 2011).

Some scholars have distinguished between stories, and systems capable of generating stories: Koenitz calls the latter protostories (2010), Montfort identifies parser-based interactive fictions as systems for producing narratives rather than narratives themselves (Montfort, 2003, p. 25-6), and in prior work I have called these artifacts “story systems” to make a similar distinction (Reed, 2011b). I now find this a less useful distinction to make, and my use of “storygame” in this dissertation encompasses both kinds of systems. The broader use of the term lets the storygame lens be applied to a wider range of unique systems (including games in which the units of narrative are supplied by players at runtime, rather than from a pre-generated stock: see chapter 7).

Storygames are also not inherently digital: the famous *Choose Your Own Adventure* gamebooks meet all requirements of our definition. However, much of this dissertation discusses improving digital storygames through new design and computational techniques, and therefore most of the games discussed (especially in the first two of the three centerpiece chapters) will be digital.

1.1.3 Expressive Input

Storygames best demonstrate their interesting qualities when the player can become actively involved in an exploration and interrogation of both their ludic and narrative components. In *Expressive Processing* (2009, p. 3-5), Wardrip-Fruin refers to the “surface” of a system (the parts the player directly sees and interacts with) as an abstracted interface into the internal data and processes within it. The word “expressive” is used in discussing a system’s internals to indicate that authorial intent can be recognized in a set of encoded processes, either through the output of those processes or a study of their internal design. An expressive process is one through which an author’s intention can be seen.⁹

We might reuse this term to also characterize the interface to a system, and say that a storygame (or any system) has **expressive input** if the user can express their distinctive intentions through it. I mean distinct in much the same sense we expect an expressive process to demonstrate a distinct authorial intent: not one of three or four possible intents, but an intent specific to that particular author. In the same sense, expressive input implies the player’s space of possible interactions is not easily enumerable: is large enough that the player might feel ownership, discovery, or surprise as they consider possible ways of interacting with the system.

To illustrate, imagine two character creation systems for a role-playing game.

The first, Class Picker, offers players a choice between five character types, including

⁹An expressive process might also offer unintended insights into the author of a work, such as cultural biases or contradictions of stated intentions: authors might express more than they realize. I’ll generally use the term only to refer to deliberate expressivity.



Figure 1.2: The character creator in *Mass Effect: Andromeda* (2017), which provides expressive input (regardless of whether the system uses this input to drive any expressive processes).

Rogue and Wizard. The second, Skill Creator, lets players assign a pool of stat points among five possible skills, including Sneaking and Magic. The second system offers expressive input while the first does not. Skill Creator's expressive input, unlike that of its competitor, can be characterized as follows:

1. The range of outcomes implied by the possible inputs may not be immediately obvious to the player, allowing for “discovery” of at first unconsidered options: for instance, equally splitting points between Magic and Sneaking to make a mystical assassin. Some possible inputs might even surprise the system’s designer (such as splitting all point equally to make a generalist).
2. While the set of possible inputs is technically finite and enumerable (each possible distribution of points), it would not be practical or useful to present those possible inputs to the player as a comprehensive list of options.

We might therefore say that systems like Photoshop, *Adventure*, or the *Spore Creature Creator* (2008) have expressive input, and systems like *Pong* (1972), *The Cave of Time* (Choose Your Own Adventure gamebook; 1979), or a phone tree, all with easily enumerable inputs and little support for surprising, discovered actions, do not. However, it's useful to note that this is a subjective distinction. Two people might disagree about, for instance, whether or not the game of Go has expressive input. But generally speaking we can use the concept of expressive input as another lens, distinguishing systems whose inputs tend to offer potential for discovery (for the player) and surprise (for the designer) from those which tend not to.

Some additional points:

1. Expressive input does not imply expressive processes or interesting output. The storygame attached to the expressive Skill Creator might not implement all of the listed skills, or perhaps contains a bug (or deliberate behavior) that makes all outcomes determined by random chance instead of skill checks. This does not change the perceived expressiveness of the system's input.¹⁰
2. Expressiveness is often created through multiplicative inputs (such as a verb plus a noun, or a number plus a skill).¹¹
3. Expressive input is not inherently superior to alternatives: indeed, it has many weaknesses, some of which will be explored in our discussion of adventure games (chapter 2).

¹⁰We could make a distinction between a surface's *perceived expressiveness* and *functional expressiveness*, depending on the extent to which the distinctness of the player's input is honored by the system; but in practice these differences can be discussed more directly through the language of narrative logics introduced below (1.2.6).

¹¹Single-input expressiveness is also possible, as in a color palette selector, or the game *Scribblenauts* (2009) in which thousands of items can be created by typing in different words, or even perhaps in *Myst* (1993) where the number of potential objects to click on in a large set of linked screens feels too large to easily enumerate.

Storygames both with and without expressive input will be considered throughout this document, but most of the new systems proposed include it. Interactive narratives in which the player’s input space is large enough for discovery and surprising action have the greatest potential to create an interesting conversation between expressive processes (from designers) and expressive input (from players), so these are the kinds of storygames to which I focus most of my attention.

1.1.4 Why Fiction?

Finally, after this discussion of the game-like aspects of storygames, it’s worth pausing for a moment to focus on the story side, too, and why it’s important. As with other popular media, the vast majority of released games with a narrative component continue in the tradition of reading for immersion (Ryan, 2001b) that rose to prominence along with the novel, had a cultural and popular heyday in the nineteenth century, and declined among the literary avant-garde, while remaining commercially popular, during the twentieth. Some games scholars (most famously Murray, 1997) see this not just as an artifact of popular culture’s continuing escapist fascination for these kinds of stories, but inherent to the spread of technology into the way we tell them: “It being the goal of authors of immersive fiction to involve their readers in their stories as much as possible, giving them some sort of agency within the storyworld was a natural development” (Maher, 2008).¹² Others have argued, however, that if interactive narratives are merely

¹²See also the fascinating argument in Peterson (2012) for the inevitability of fantasy as the primary genre for both tabletop roleplaying games and eventually their more immersive digital descendants, as extensions of the genre’s foundational convention: the transportation, for a limited time and usually at a profit, of regular people to fantastical places.

systems for generating traditional, linear stories, they are not taking full advantage of the birth of a genuinely non-linear medium (Bernstein, 2009). Much of the academic hypertext fiction tradition therefore embraces modernist and postmodernist schools of narrative that de-emphasize immersion.

Immersive fiction remains extremely popular outside literary academia, of course, and the most widely played playable media works are immersive games. If we want to keep the focus on learning from completed and released experiences, we will find a far larger data source in immersive storygames than in other forms of playable media. But more significantly, I believe there is a fundamental affinity between play and immersive fiction that belies the relationship's competitive reputation. Gingold (2016), for instance, identifies several foundational qualities of play, including that it is intrinsic (players find it absorbing or pleasurable), and has variability under constraints, which enables agency (the feeling of having a transformational effect). Play and stories are engrossing, empowering, and sometimes transforming. A satisfying traversal of a storygame is a rewarding experience in part because of the shared and linked pleasures of narrative immersion and ludic accomplishment. Works that center this cooperative linkage are worthy of study.

It's also worth noting that traversing a storygame does not have to be satisfying in the sense we usually associate with words like *play* or *fun*. Storygames can explore tragic, painful, or challenging themes. While some theorists have argued that interactive tragedies are inherently unworkable since players want their characters to come out on top (Ryan, 2001a; Bernstein and Greco, 2004), I have previously written about existing

work that effectively challenges this claim (2011a), including creating such a project of my own (*maybe make some change*, 2011); and later in this dissertation I will discuss the interactive tragedy storygame *Polaris* (7.3.3). Frustration and lack of agency can be employed just as effectively in creating effective interactive experiences as flow and power fantasies. But it's certainly true that designers to date have far fewer examples, tools, and techniques to create these kinds of effects, and game players have much less experience with how to engage with a game that asks for uncomfortable emotions. It's interesting to note that many such games discussed herein, including *Polaris*, or games by the designer Squinky (7.4.2), involve ritualistic acts at the beginning of the experience to draw the player into the “magic circle” of the game world (Huizinga, 1950), a technique that theatre has used for centuries to draw audiences into something potentially tragic or otherworldly (Turner, 1982).

Finally, by centering “fiction” I do not mean to exclude interactive works of non-fiction, such as interactive documentary in film, VR and AR art that engages with real-world issues, explorable recreations of historical spaces, game autobiographies, newsgames, simulation and edutainment, or database fiction. Transforming a true event into an interactive experience inevitably makes it both a story and a system; many these artifacts can be discussed as storygames as productively as those with strictly fictional foundations.

1.2 A Framework for Storygame Discussion

1.2.1 The Slipperiness of Genre

Game genre is a frustratingly inexact concept. Games suffer from “genre muddle” (Adams, 2009), with a number of sometimes contradictory spectra along which we might choose to categorize them. Adams identifies four such qualifiers that have been used for genre classification in games, including audience (capturing distinctions such as casual versus hardcore gamers), theme, and purpose. Game genres have been defined through how the player interacts with the game (Fernández-Vara, 2009) or as a consequence of the interface and skills required to play (Plotkin, 2011) or by a shared collection of core mechanics (Costikyan, 2005). Games marketing has its own notions of genre:

nearly all commercial narrative games, for instance, are lumped into a couple of inexact categories like *adventure* or *RPG* (Figure 1.3). This seemingly arbitrary bucketing has a real effect on which games we consider similar and different. For instance, despite the fact that sports games have made significant advances in fields like social simulation, narration of dynamic stories, and drama management (Garcin, 2013), these games are rarely studied alongside other kinds of narrative games and much of the design work in

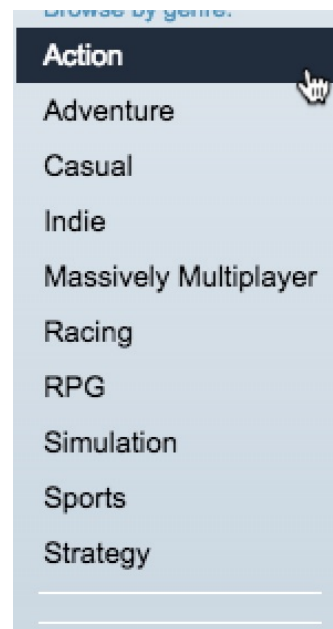


Figure 1.3: The game genres, according to games distribution service Steam.

these two communities is unknown to each other.¹³ Given the problems with genre, how can we make useful distinctions between different styles of storygame?

On a superficial level, we can bump the problem up to a higher level by adapting the literary distinction between *modes*, such as the comic, ironic, or didactic mode. Literary modes are a “critical term usually identifying a broad but identifiable literary method, mood, or manner that is not tied exclusively to a particular form or genre” (Univ. of Richmond Writing Center, 2010). Interpreting a work as an example of a particular mode can offer insights into the intentions of its creator and tools for understanding its aims and how it goes about advancing them, without being quite as categorical or imperfectly precise as genre.

For a more useful solution, however, we will need to examine layers of abstraction around what actually happens in a game: the actions the player takes and the responses the system makes in return. In the rest of this section I will build up a vocabulary for making these comparisons, culminating with useful language for comparing the ways very different storygames handle making their narratives interactive.

1.2.2 Inputs, Events, and Actions

On what level is it useful to discuss a player’s input to a storygame? Computer systems and their interfaces are often productively seen as a series of abstractions at higher and higher levels (Agre, 1997). Following this line, we might break a player’s interaction into three levels of abstraction: the raw **input**, the processed **event**, and the

¹³Sports games in the context of social simulation are briefly discussed in (5.3.4.5).

resultant **action**. The input represents a physical unit of interaction from the player, such as a mouse click. The event represents that interaction in a form the system has both validated and is prepared to accept, such as a click event on a button object. Finally, the resulting action is the process triggered by an event, such as the action of dismissing a notification.¹⁴ Specific actions (*e.g.* dismissing notification 173) are then understood as instantiations of affordances, or capabilities, that the system provides to the user, such as the affordance of dismissing notifications. In most cases the action/affordance level is the lowest that needs to be considered, although in certain cases considering how intent flows between these levels can be productive.¹⁵

1.2.3 Mechanics and Dynamics

A set of affordances, along with the game rules and UI elements that instantiate and react to them, can be called a **mechanic** (Hunicke et al., 2004). For instance,

¹⁴A series of inputs might also be abstracted into a single event (such as a double-click), and multiple events might in turn be abstracted into a single action (such as selecting a group of units and clicking a point to move them to in a real-time strategy game).

¹⁵For instance, if the player experiences friction in translating their intent through these levels, it can result in a play experience that feels frustrating, unintuitive, or broken.

Consider the differences between choice-based storygames (where the player selects potential actions from a list) and parser-based games like *Adventure*. In a particular implementation of the former, players might have a single input (clicking) which can either fail (if the click is not on a choice button) or succeed (if it is). A successful click becomes an event (activating a button) which directly becomes an action (selecting a choice). This supports a single affordance of “choosing what to do.” The connections between the levels are abundantly clear.

Now consider the parser. Inputs (pressing letters on the keyboard) are aggregated straightforwardly enough into events (submitting a command like `GO TO THE BUILDING`), but it’s not always clear to the player which events will produce successful actions. For instance, in some implementations of *Adventure* the mentioned command might become a valid action, while in others it would be rejected with an error. Furthermore, it’s not always clear how actions relate to affordances: in some parser games, the actions of “looking behind” and “looking under” might produce identical results, while others might treat them as separate and unrelated affordances.

These speed bumps between different levels of representation can be a different way of understanding how the player’s experience with a system’s input can shape their perception of it.

input	event	action	mechanic	dynamic
mouse click at coordinates 143, 74	Click button (OK)	Dismiss notification	-	-
key pressed 'y'	'YES' event	Dismiss notification	-	-
raw audio file	Voice command "Close window"	Dismiss notification	-	-
series of skeletal keyframes	Gesture "wave left"	Dismiss notification	-	-
series of key presses 'N', 'O', 'R', 'T', 'H', 'enter'	player command "north"	the action of going north	movement	exploration and mapping
series of key presses ...	player command "drop sword"	the action of dropping the sword	object manipulation, limited inventory	inventory management
touch at coordinates x, y	Option #3 selected	Choosing dialog option "You ask far too much."	Conversation tree	lawnmowing
end drag at coordinates x, y	Drop light 4 on socket 9	Activating symbol "romance"	Switching on story fragments	exploring possible story configurations

Table 1.1: Examples of player interaction at different level of abstraction.

a game might have a movement mechanic that encompasses the separate affordances of walking and running, as well as internal variables setting relative speeds of various game characters, movement animations, and a fatigue bar visible to the player. We can also speak of the **dynamics** arising from a mechanic or the interplay between several mechanics, such as players conserving their fatigue points to escape enemies in short quick bursts.

The stack from input to dynamic and some examples can be seen in Table 1.1.

1.2.4 Operational Logics

A player often interprets meaning from a playable system by observing its behavior across these different levels. One such window of interpretation is the final layer

of Hunicke’s framework, aesthetics, which is seen as arising from dynamics as perceived by the player. However, a different approach to understanding how we holistically interpret a system is to consider it from the system’s perspective. The term **operational logic** (originating in Wardrip-Fruin, 2005) has been used to describe an authoring strategy “for specifying the behaviors a system must exhibit in order to be understood as representing a specified domain to a specified audience” (Mateas and Wardrip-Fruin, 2009).

Encoding such patterns requires a system to capture input, code to process it, output to express its result, and a unified design connecting these components together such that they seem to represent a recognizable phenomenon. While mechanics are largely centered on the player’s actions and systems supporting them, operational logics also include perhaps hidden internal processes (Wardrip-Fruin, 2009). Operational logics are thus another framework for weaving together the layers of abstraction in a storygame system, its external surface, and its internal processes.

1.2.5 Narrative Mechanics

When discussing storygames in particular, it can become useful to speak of the subsets of mechanics, dynamics, and operational logics that informs how the story is presented to and manipulated by the player. **Narrative mechanics** are those affordances that directly concern how the player selects and navigates the lexia in a storygame. Similarly, **narrative dynamics** are the patterns of behavior that arise from these mechanics.

There are a surprisingly small number of well-established narrative mechanics.

While storygames often include mechanics common to many styles of game (such as for movement or inventory management), I can think of only a handful of mechanics that specifically operationalize narrative content:

1. Explicit story choice (selecting one of several alternatives to receive a related piece of story: dialogue trees, choice graphs)
2. Puzzle-solving (performing an at-first unclear sequence of actions reveals story)
3. Story dumps (touching something in the world reveals story: clicking on a quest-giver, activating an audio diary, opening a journal entry)
4. Character customization (building or customizing your avatar: choosing a class, assigning skill points)

There are more examples of narrative dynamics, due to the way different implementations of narrative mechanics can interact with each other and with non-narrative mechanics. For example, an explicit story choice mechanic that doesn't move the story forward, combined with a design where certain choices give positive benefits, can lead to the dynamic called *lawnmowing*, where the player feels the need to exhaust every possible option in a given list before continuing (Short, 2009). Story dumps with no ludic consequences for activating them (such as audio logs scattered around a level) can lead to a dynamic where the player perceives the story as optional and loses motivation to seek out and activate lexia.

A common source of tension in negative narrative dynamics is a mismatch in expressiveness between the narrative mechanics and the systems supporting them,

including the way the game’s operational logics handle narrative processes. We will turn to a discussion of these internal processes next.

1.2.6 Narrative Logics

While narrative mechanics represent the kinds of things a player can do with the story in a game, consider as well the kinds of things the system can do: the internal processes that act on player input to process and return narrative lexia or generated output. These are the **narrative logics** of a storygame. In the same way narrative mechanics are the subset of player affordances connected to story, narrative logics are the subset of operational logics connected to how narrative actions are processed, and how narrative output that seems connected to those actions is produced.

Traditional media does not really have narrative logics, although to illustrate the point I could say a digital implementation of a book or film has a narrative logic consisting of the function `next()`: display the next page, or show the next frame. Story content in linear media is conceptually a linked list, a one-dimensional line segment with progression from a first lexia through to a final lexia. We might interpret a rewind button as an action that invokes the narrative logic *display the story in reverse order and more quickly*. Within the context of storygames, we can speak of a narrative logic like choice graphs (discussed in 1.2.7) where the player’s selection of a choice reveals the narrative content linked to that choice. Several other examples of narrative logics include:

1. Undirected Graphs - edges are not necessarily directed; the player can freely

explore a fixed graph of lexia (perhaps representing a series of rooms or a set of connected memories)

2. Triggered Content - Lexia are triggered when a particular game state arises (such as time of day, or a certain number of points being reached)
3. Random triggered content - Lexia are triggered at random (wandering monsters; a randomly selected villain in a murder mystery)
4. Managed Content - The system attempts to alter the order, details, or availability of lexia to push the player towards a more ideal (author-defined) experience

Identifying the narrative logics in a work provides a language for discussing how a player might perceive that they are taking part in a story responding to their actions. Critically, it lets us troubleshoot if this process breaks down for a particular game or style of game. We can attempt to identify the source of the breakdown: is it in narrative mechanics whose expressiveness does not match the system's narrative logics? Is it a mismatch between the player's understanding of how their input is translated into actions? In the next chapter, we'll use this framework to do a detailed study of adventure games, discovering that their narrative logics were often at odds with how players perceived their affordances.

Narrative logics are also a lens to talk more critically about how a storygame's design and code, not just its lexia, affects the kind of stories it can tell. The three primary chapters ahead are each centered around a different kind of design for a new narrative logic.

First, however, let's briefly analyze a specific existing narrative logic to demon-

strate how this process can provide useful insights about games which employ it.

1.2.7 The Choice Graph: A Standard Narrative Logic

Among the most common narrative logics used by interactive stories is the *choice graph*, a directed graph with nodes representing lexia and labeled edges representing explicit choices offered to the player. Typically in storygames using this logic a traversal takes the player (and usually also the player character) from a starting node to a leaf node with no outgoing edges, representing a conclusion (sometimes an abrupt one) to the story. Moving forward in the graph generally corresponds to moving forward in time through one branch of a multicursal plot.

The procedural aspects of this logic are straightforward, which we might guess by the fact that one can instantiate it in a printed book. An action for choosing options enables a mechanic of selecting choices. A choice graph system almost always maps a specific choice to a specific target lexia, and handles moving the player along the correct edge to that target.¹⁶ While the graph structure might be quite complex, it seldom or never changes, and the algorithm driving its navigation is straightforward: move along the selected edge to the connected node.

Choice graphs show up in many contexts in storygames. While they can be connected to the specific narrative mechanic of explicit story choice, they might also be behind other mechanics. In a *conversation tree*, this narrative logic is applied to

¹⁶In book form, the label of the edge, the node it connects to, and the procedure for translating an action into the next lexia are encoded in a single human-readable sentence, like *If you want to attack the bear, turn to page 93*. The edge label is *attack the bear*, the destination is *93*, and the algorithm is *turn to page n*.

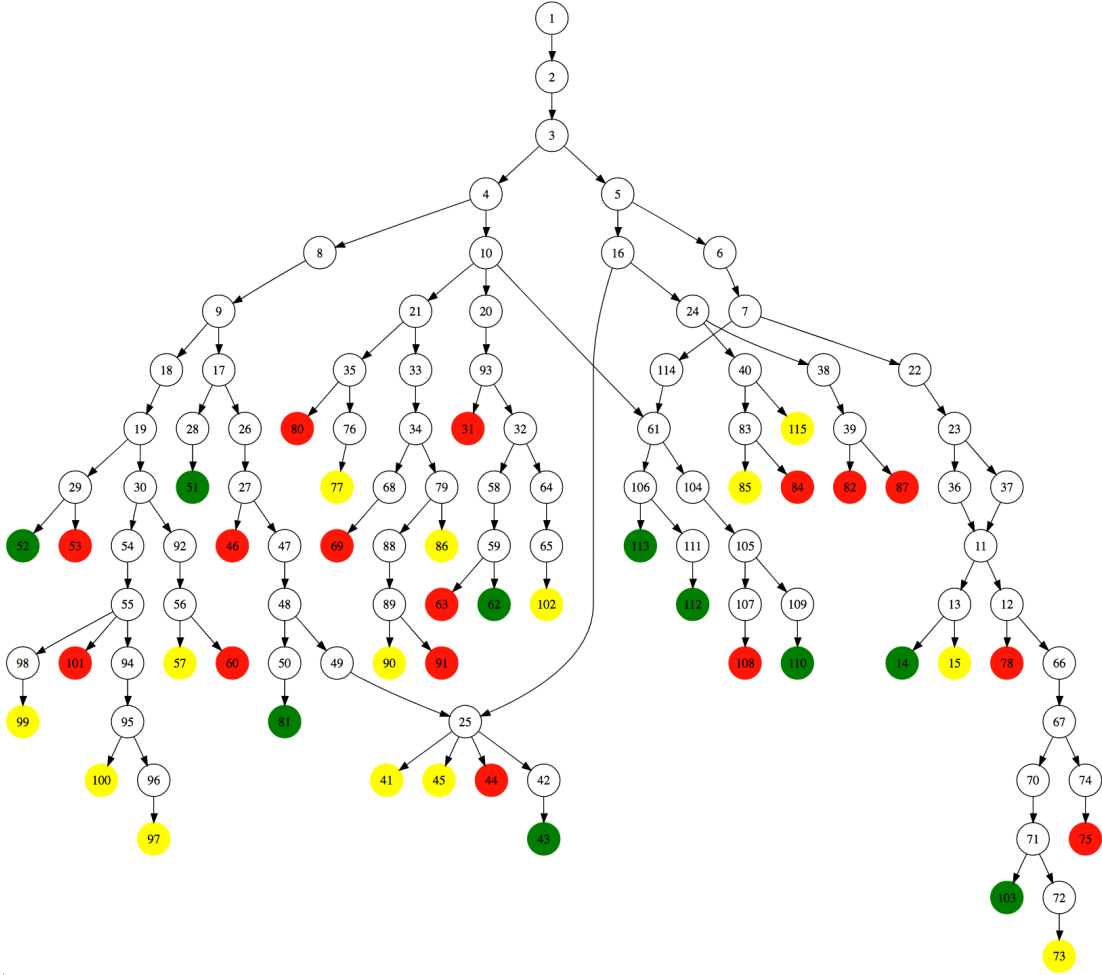


Figure 1.4: Choice graph from *The Cave of Time, Choose Your Own Adventure #1*. The player begins a traversal at node 1, making a choice at each node with outgoing edges, until they reach a colored ending node and complete a traversal. Graph courtesy “The Outspaced Shrine” (outspaced.fightingfantasy.net)

a subset of a larger game (which may be using the same or different narrative logics) that represents a single conversation. After each story lexia (a sentence or two from the NPC), the player is given a series of choices of possible things for the player character to say next. Each one corresponds with a connecting node on the graph. Reaching certain nodes, or ending at specific terminal nodes, might affect the operation of another narrative logic driving overall game progression (such as angering a character so much they decide to fight you, or triggering the beginning of a new quest). One of those higher-level logics might also be a choice graph representing several possible endings, as in the endgame of *Myst*: within this instantiation of the choice graph logic, the player spends most of the game in the first node, then makes a choice near the game’s ending (perhaps on a higher level than a single action) to move into one of the connected and terminal leaf nodes.

If the directed restriction on choice graphs is loosened, we can find other places where story games make use of this narrative logic to sequence content. Movement through space is represented by an undirected graph in text-based interactive fictions (rooms), graphical adventure games (screens), or point-and-click exploration games like *Myst* (viewpoints). An interesting thematic application of this logic is the storygame *The (Former) General In His Labyrinth* (2008), in which the player character is a Middle Eastern dictator wandering the rooms of his mansion, caught in endless cycles of political feuding and social isolation. The narrative content as well as the descriptions of people and locales repeats as you revisit each room, creating a dynamic of hopelessness as the player discovers the bounds of the player character’s physical, as well as his social and

emotional, domain.

Another variation on choice graphs is to add a blackboard¹⁷ that can be read from and written to, preserving arbitrary state as the graph is traversed, and consulting that state to determine which edges to reveal to the player. One implementation of this used in some hypertext fictions is the guard field (Joyce, 1991), preventing access to a lexia until a certain condition has been met. The commercial storygame company Choice of Games (whose games exclusively use explicit story choices driving directed choice graphs) advises authors to create five to six “character stats” which are adjusted up and down by visiting certain nodes; later in the story, these stats gate additional choices that can unlock different lexia or endings, an authoring technique they call “delayed branching” (Fabulich, 2011). Game state can also be used to alter how text within a templated lexia is displayed.

This addition of state can change the dynamics of choice graphs, giving players a stronger feeling that their choices matter. While one player might reach the same lexia as another, they might have done so via a different path and thus accumulated a different state, which in turn might alter the text they’re seeing or the choices they’re provided with. Adding state can make the explicit story choice mechanic more expressive: multiple player actions can aggregate into a multiplicative expression of more complex intent (such as mostly being honorable but sometimes being sneaky) represented by the

¹⁷A term of art in computer science where code can write to and read from a shared space by storing or retrieving data associated with a particular key: for instance, we might write to the blackboard that the key `time_of_day` corresponds to `evening`, and be able to access that value given that key later. This is more or less like having globally available variables, except that keys do not need to be defined in advance (so individual elements can feel free to write to newly invented keys).

state and responded to by the game. A slight change to narrative logics can produce more satisfying play, by providing a game whose output feels more expressive, and a story that feels more reactive.

This analysis can also more precisely reveal the limitations of choice graphs. It's clear from their static structure that, state preservation or no, the system will never be able to respond to states not considered by the designer. Adding code to alter lexia or available choices creates the illusion of a dynamic graph, but the reality is simply that some of the authored lexia are hidden on any given traversal.

We can also make observations about how modifications to the logic have ludic consequences. Comparing directed and undirected choice graphs, it's clear that with the former *every action moves the player closer to an ending*, while in the latter *an ending might never be reached*. This might have profound design implications. For example, the difficulty of adventure games (which use undirected choice graphs to represent a spatial map) may in part stem from the fact that the player can wander indefinitely without ever reaching a node that progresses the story. Furthermore, this is true even for a game without puzzles: the structure of the logic itself implies it. This structural insight is used deliberately in *The Former General*, but probably unintentionally in classic adventure games. Through this lens we can also see a logics perspective on the difference in theme and content between the straightforward, action-heavy nature of gamebooks and the postmodern restlessness of hypertext fiction, or the aesthetic of hopelessness in a relentlessly railroaded¹⁸ game like *Heavy Rain* (2010) versus the sense

¹⁸A term for a storygame that moves the player from one lexia to another with few or no opportunities to make choices: that is, a directed graph where most nodes have only a single outward edge.

of discovery and adventure evoked by one's initial encounters with the open world in a *King's Quest*.

What are some takeaways from this analysis of a particular narrative logic?

1. Many narrative logics might be operating within a single storygame, including even the same logic representing different levels of abstraction within the narrative.
2. Even though a specific storygame will have a specific implementation of a narrative logic, we can identify useful commonalities and recurring patterns in games that share common structures.
3. When the player's perception of their narrative affordances (affected perhaps by the lexia they've seen) does not align with a game's narrative logics, negative dynamics can result.

This discussion of choice graphs brings together the elements of the framework and demonstrates how it can be used to make holistic observations about how different levels of a story game are interrelated and can work together or against each other. I will now turn this framework to the analysis of an existing storygame genre, and use it to look for insights into how it succeeded and why it failed.

Chapter 2

The Adventure Game: An Existing Storygame Mode

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.

Dave Lebling, co-creator of Zork (1979)

Before proposing new kinds of storygame, let's begin by turning the framework defined in the last chapter on an existing one: the adventure game. Dominant through the 1980s and commercially viable until the late 1990s, the adventure game is still among the most famous genres of storygame despite two decades of commercial stagnation, since it is one of the few styles of game to foundationally foreground story. Adventure games offer a unique pleasure of discovery that merges both ludic and narrative planes, the “eureka story.” By using our framework, I will show how this dynamic is enabled. However, it will also reveal that adventure games have a fatal flaw which in part led to their downfall: a trio of cornerstones (exploration, puzzles, and story) which are

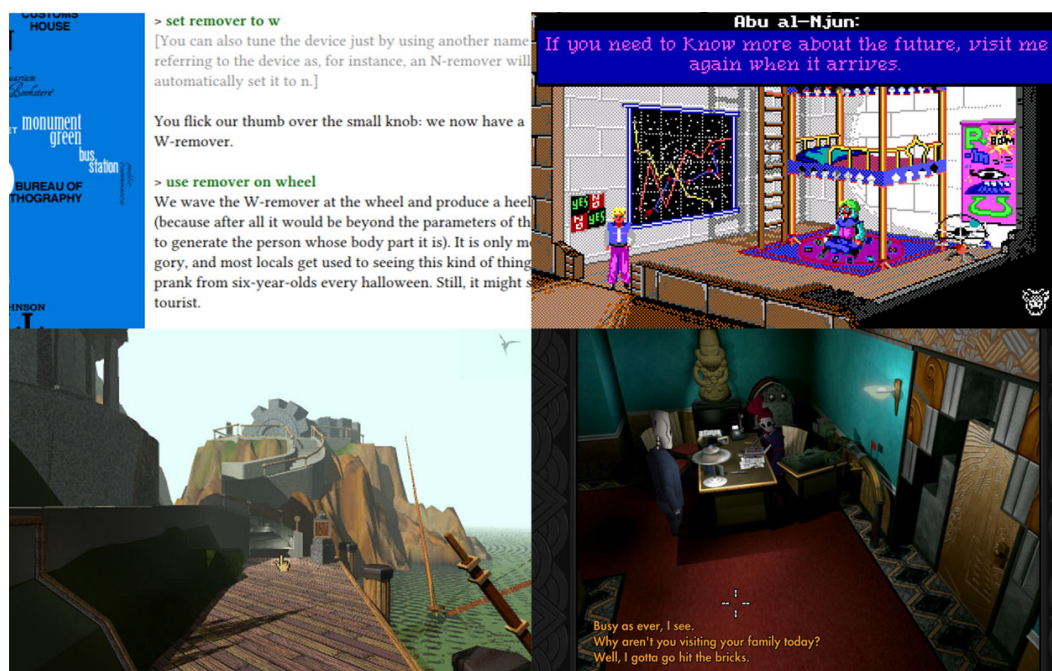


Figure 2.1: Various storygames that share the core design features of adventure games, including (clockwise from top left) *Counterfeit Monkey* (2012), *Quest For Glory II* (1990), *Grim Fandango* (1998), and *Myst* (1993).

connected in such a way that having narrative logics effectively joining all three is extremely difficult. A deep understanding of this seemingly-dead genre can provide surprising insights into contemporary games that descend from its traditions, however. I will study three successful contemporary games (*Firewatch*, *The Witness*, and *Her Story*) that each remove a different foundational component of the genre to focus on the logics that amplify the connection between the remaining two. Finally, I will suggest that an alternative possibility to preserve the story-driven pleasures of adventure games can be found in new technological and design advances, which will point the way towards the next three chapters of the dissertation.

While scholars and practitioners have often used “adventure game” to refer

exclusively to graphical adventures of a specific lineage, we will see in our analysis that we can productively define the genre through its narrative logics, which are largely unchanged across games such as *Zork*, *Myst*, *Grim Fandango*, and *Quest For Glory II*. All are storygames featuring diegetic puzzles, exploration and experimentation by a player-controlled avatar, and a story graph in which nearly every node must be visited in a complete traversal (as opposed to choice-based narratives, in which only a fraction of all possible nodes is generally seen). We will begin by assembling this structural definition from its historical building blocks.

2.1 Definition

As we will do in future chapters, let's begin by reconsidering our definition of a storygame, which we've said is:

- a playable system, with
- units of narrative, where
- the understanding of both, and the relationship between them, is required for a satisfying traversal.

As described in 1.1.2.1, it's clear that a traditional adventure game like the original *Adventure* falls within this definition. What I'd next like to do is determine how adventure games act as a specific subcategory of storygame: that is, what qualities do all adventure games share in addition to those listed above that other kinds of storygames do not. I posit that adventure games enact three core dynamics on top of a storygame's

basic structure:

- solving puzzles
- exploring a world
- starring in a complete story

In a good adventure game, these three dynamics are carefully interconnected, as we'll discuss further below. But first, let's consider how we arrived at these three features by considering how definitions of adventure games emerged and evolved throughout their brief history.

The evolution of adventure games can be traced through a lineage of experimentation, stability, and stereotypes. Exploding into the consciousness of computer users with the rapid distribution and success of Crowther and Woods' *Adventure* in 1977, the genre helped spawn the computer game industry and drove some of its first decade's best-selling titles, weathering rapid and dramatic upgrades to the multimedia capabilities of home computers.¹ The genre declined in popularity slowly throughout the 1990s,² despite the 1993 debut of *Myst*, its best-selling entry. By the end of the decade,

¹Curiously, adventure games are largely a computer phenomenon, not a console or arcade one. Home computers were originally the only platform that offered both expressive input (a keyboard or mouse) and storage media to allow save games, so progress in a complex story could be saved and resumed later; both factors may have been a part of why adventure games flourished on this platform (Salter, 2014, p. 40).

²Historical sales figures for games are difficult to track, but the first adventure game to sell a million copies was released in 1980 (*Zork*) and the last, arguably, in 1997 (*Myst*'s sequel, *Riven*). The Software Publisher's Association's "CoDie" awards had a category for adventure games through 1997 (and the following year had to give *Riven* a special "Best Overall Multimedia Presentation" award instead). On the other hand, Golden Joystick (a British popular choice award) had removed their "Adventure Game of the Year" category back in 1989, while the magazine *Computer Gaming World* kept theirs through at least 2005, demonstrating the wide range of time over which the genre declined in commercial and popular success.

the commercial failures of titles like *Grim Fandango* (1998) and *Gabriel Knight 3* (1999) had marked the end of adventure games as a mainstream genre, though smaller studios and hobbyists have kept them alive ever since (Nelson, 1995; Salter, 2014).

The reason for the decline and fall of the adventure game will be discussed more thoroughly below. First, however, we need to define more precisely what exactly we mean when employing this term. Historically speaking, a definition of adventure game proved rather difficult to pin down, and is still the subject of some debate.

1977’s *Adventure*, as a singular phenomenon, needed no descriptor at the time more than just “game” or “program,” the two descriptive words used in one of its first mentions in a printed book (Kidder, 1981). As it began to spawn imitators that flooded the early Arpanet, it was perhaps natural that these were called “Adventures”³ or, eventually, “adventure games,” although this terminology took a while to become standardized. When *Zork* appeared in 1979 as one of the first commercially successful *Adventure* imitators, its creators labeled the genre somewhat clumsily as “Computerized Fantasy Simulation,” tying its lineage closely to tabletop roleplaying games like *Dungeons & Dragons* as well their digital imitators like *Rogue*.⁴ *Zork*’s creators stated the defining characteristics of their new genre as follows:

First, the object of the game is usually to collect treasure, and this may be done only by solving problems. . . Second, a great deal of the enjoyment of

³For example, this was the term used by early innovator Scott Adams to describe and market his products (Maher, 2011).

⁴These latter games, with a narrative component but more focus on statistics, skill, and sometimes reflexes, continue to this day as “roguelikes” but also spawned their own family tree labeled “computer role-playing games” or CRPGs. This genre mitosis took place over some time, however. In 1985’s *The Book of Adventure Games*, Kim Schuette still identified adventure games as a parent genre containing both “puzzle adventures”—what we think of as adventure games today—and “fantasy adventures” like computer role-playing games.

such games is derived by probing their responses in a sort of informal Turing test: ‘I wonder what it will say if I do this?’ The players and designers delight in clever (or unexpected) responses to otherwise useless actions. (Lebling et al., 1979)

Solving problems, usually under the name of *puzzles*, is by far the most common feature identified as core to adventure games, appearing in every definition we’ll consider here. Designer Al Lowe has suggested (1999) this dynamic perhaps sprouted from the often inscrutable puzzle of dealing with early computers in the first place: staring at a command prompt trying to figure out what to do next was perhaps only marginally less frustrating in *Zork* than Unix.

One difference is that puzzles, at least the good ones, had narrative context. A much later definition of adventure games clarifies that their puzzles should be “integrated in the fictional world” (Fernández-Vara, 2009, p. 13), although this concept was nascent even in Lebling’s prototypical definition: the player of *Zork* isn’t solving puzzles purely for the fun of it, they’re doing it to “collect treasure.” Indeed, as the genre matured, well-integrated puzzles began to be seen as part of the art of adventure games, such that those in *The Seventh Guest* (1993), with its house full of arbitrary brain teasers, could be derogatorily compared with *Riven* (1997) which integrated its puzzles more seamlessly into the fictional world (Barba, 1997). This is only the first of many overlaps between the three pillars of adventure to be teased out below.

Another point about puzzles as a central mechanic is that they are solved through wits, not reflexes. Another late-period definition of adventure games says they have “a reduced emphasis on combat or action elements.” This is the core of the split



Figure 2.2: A puzzle integrated into the story world in *Riven*. This children’s toy, found in an abandoned classroom, both provides clues to help you decipher the D’ni numbering system, and continues a story of the cruel nature of Gehn’s control over the inhabitants.

between adventure games and computer role-playing games, despite the two genres sharing other attributes (Rollings and Adams, 2003).

By 1986, with hundreds instead of a handful of examples, definitions had grown slightly broader: many adventure games had appeared based on ideas other than treasure hunts, for instance. One of the first published guides to creating one’s own adventure games defines them as “in essence, a puzzle or series of puzzles” (Hartnell, 1984, p. xi), but goes on to clarify that these puzzles must be placed in a consistent environment which the player must gain mastery over, such as through producing a map. The notion of a consistent environment that must be explored is a second pillar of adventure games,

along with the related notion that there must be someone to do the exploring. Another mid-eighties guide to adventure games defined the genre as:

one in which the computer provides an alter ego for you, the game player. . . . The challenge usually involves solving many puzzles in pursuit of the final reward. . . . The search for surprises and a little humor in unexpected places make up another key factor. . . . Some of the pleasure of these games derives from succeeding after repeated failure. (Schuette, 1985, p. 1)

Puzzles appear again in this definition (along with the clarification that they should not be trivially easy), but new is the notion of the alter ego, a specific character who is solving the puzzles and pursuing a reward. While some adventure games conflate the notion of player and player character through use of the second-person voice or a first person camera, the genre became best known for games where the avatar was visible on screen as a particular, personified character (Salter, 2014, p. 4).

A disconnect between the consistency of the environment and the conceit that the player character operates within it can be detected in this longer definition from the authoritative-sounding *Encyclopedia of Microcomputers* (vol. 3, Kent and Williams, 1989), in which one might also detect some hints of the jaded frustration that would later send the genre to an early grave:

[T]he player is placed into what appears to be a realistic fantasy world. The objectives are rarely stated so the player must not only figure out what to do but what the objective of the game is. The player can move from location to location, pick up and drop objects, and take various kinds of actions. Objects include (1) “treasures,” which increase the players [sic] score, (2) weapons, which the player can use to fight the various opponents of the adventure game, (3) literal or figurative keys, which are important in solving puzzles of the game, (4) objects containing clues, and (5) some objects that are totally useless.

This definition (perhaps somewhat passive-aggressively) identifies several frustrations that can arise when the dynamics of exploration are poorly integrated with those of story and environment. Useless objects and unclear objectives are signs that the player’s exploration—their experimentation within the environment—has been unproductive. But object manipulation (successful and otherwise) as well as movement show up in adventure game definitions precisely because they enable the key dynamic of exploration and experimentation, which is necessary for a player to understand enough about the world to solve its situated puzzles.

For experimentation to be effective, the world must be consistent (as Hartnell pointed out), and another word for a consistent world that can be acted upon is a simulation.⁵ Another corollary of experimentation within a simulated world is that the experiments must be necessary: that is, the solution must not be obvious. Recall Schutte’s definition from 1985 stating that “the pleasure of these games derives from succeeding after repeated failure.” More recently, Douglass (2007) notes that these games⁶ can be characterized as a continuous process of interrogation: asking questions

⁵This may seem a problematic word to use in light of Aarseth’s distinction between simulated and fictional game elements, discussed later in 5.2. Briefly, if an object labeled “fish” can only be used to solve one puzzle in an adventure game and not in any other contexts one might use a fish, can we really say the game has simulated a fish? Indeed, Aarseth has explicitly argued that adventure games are not simulations. However, Fernández-Vara (2011) disagrees:

It is the simulative aspects that provide adventure games with one of their defining characteristics, the exploration of action and space. If the world were not simulated, players could not try picking things up, opening or closing doors, or talk[ing] to other characters. The fact that there is a sequence of events that must take place in a specific order, by solving the puzzles in a specific way, is different from the fictional world not being simulated. . . . The player needs to experiment in the simulation in order to learn the right thing to do.

⁶Douglass is writing specifically about command line interactive fiction, but the observation holds across graphical adventure games as well.

and receiving responses that offer a greater and greater understanding. He coins the term “frustration aesthetics” to describe how “failed” responses are actually the player’s primary means of learning about their character, the details of the world, the limits of the simulation, and the expectations of the story. For this reason adventure games almost universally have expressive input (defined in 1.1.3) enabling the player to think of possible solutions: ways in which they might manipulate their avatar in the simulated world to achieve a goal.

Interactive fiction designer Andrew Plotkin (2011, p. 64) also identifies this tension between expressive input and an open, simulated world as a core characteristic of adventure games:

Resolving this tension is in your hands, and what are your tools? The game’s description of its reality, and your understanding of it *as* reality. If you treat the words (or pixels) as interface elements of a program, you have no handhold. Any button could be the magic button. They are distinguished only by their meaning *in the game world*. You understand that an altar is an important location in a church, that a lever is an important part of a machine, that a fingerprint is an important feature of a crime scene.

The adventure game interface, in other words, is accessible only via player immersion. . . . Why does an IF game provide a simulated world? Because the player’s understanding of the world must be the primary means of determining what is possible.

So the player’s understanding of the world—arrived at through exploration and experimentation, in turn enabled by expressive input and mechanics for manipulating the world and things within it—becomes a unifying connection between the pillars of puzzle and exploration. Plotkin observes that when this marriage fails, it’s often because of poor design: puzzles that do not make narrative sense in the world and thus give the player no means to conduct experiments to learn how to solve them, or interfaces that

do not accurately carry out and report back on the intent behind the experiment. Again, adventure games live and die based on how well they resolve and connect the tensions between their pillars, and we'll shortly see that many designers made these decisions in the first place to try to connect to the third pillar: story.

Curiously, story does not appear explicitly in most early definitions of adventure games. By the time a roundtable on the decline of the genre was convened at the 1999 Game Developer Conference, however, it had risen to top billing in the definition they arrived at: “to be considered an adventure game, a game must emphasize story, include a protagonist, puzzles, and inventory objects and have a definite beginning and end” (Lowe, 1999). This teases out a qualifier that a surprising number of storygames lack, despite containing units of narrative: a complete story, with a beginning, middle, and end. Fernández-Vara (2009) describes the “ideal walkthrough” for an adventure game: the sequence of actions that will take the player character through their proscribed story (without an early death along the way). Through exploration and puzzle solving, the player attempts to discover the ideal walkthrough (and in most adventure games, a few minor variations aside, there is indeed only one).⁷ This makes the player a bit like “an actor without a script” (Fernández-Vara, 2011), hinting at another fundamental tension between the pillars. Fernández-Vara also points out the way story can strengthen these pillars, by unifying the narrative and ludic planes in a way unlike any other game genre.

In an adventure game, “advancing in the game means advancing in the story, because

⁷Contrast this with hyperfiction or choice-based games, which often have many valid paths and significantly more lexia than any player will see in a single traversal. In most adventure games, while the order may sometimes vary, and there may be a few pieces of optional side content, the player tends to have seen most of the game’s lexia by the end of a satisfying traversal.

each challenge and its solution constitute an event in the story of the game.” Each action the player takes, at least while following the ideal walkthrough, directly enacts and advances the story of the player character. The story *is* the player’s sequence of actions. Salter (2014, p. 56) notes that some adventure game strategy guides were even written as short novellas telling the story of the player character moving through a complete traversal. The ludic instructions to solve the game could be told completely in the form of a fiction.

Plotkin (2011) also observes that each adventure game puzzle tends to be unique; in the same way, the player’s narratively significant actions form unique beats in an ongoing story. Contrast this to other styles of game, where much of the story consists of repetitive actions that might be elided in other media (such as a series of battles with the same creatures) while using a smaller number of repeatable actions over and over again.

Fernández-Vara’s full definition, both recent and critically informed, makes a good final stop on our tour of adventure game definitions:

... (1) story-driven videogames, which encourage (2) exploration and (3) puzzle solving and (4) always have at least one player character. The basic interaction of adventure games is based on (5) object manipulation and spatial navigation. Their challenges usually appear in the form of concatenated puzzles, which are (6) integrated in the fictional world. (numbering mine; 2009, p. 13)

While split into six items, not three, this definition aligns nicely with our theory that adventure games rest on three core pillars of narrative dynamics:

- **solving puzzles** (3) integrated into the fiction (6)

- **exploring a world** (2) through movement and object manipulation (5)
- **starring in a complete story** (1, 4)

In distilling this definition we have also begun to tease out the interconnectedness, both positive and negative, of these pillars. Before exploring this, however, let’s consider the crucial positive dynamic that adventure games can create, which points towards why they are worth studying and saving.

2.2 Eureka Stories

A common pleasure mentioned in discussions of adventure games is the moment of revelation when a player discovers the solution to a puzzle. While non-narrative puzzle games produce similar revelations, in adventure games the moment is somewhat different: an understanding which brings together the narrative and ludic planes of a storygame, which I call a **eureka story**.⁸ This dynamic is what the genre’s three core pillars of exploration, story, and puzzles, and the mechanics that support them, are there to enable.

How does a eureka story differ from other kinds of insights? Montfort places them (in interactive fiction) in the tradition of the literary riddle, where the guesser must “awaken to a new vision of the world” (2003, p. 60); Plotkin writes about “the player’s need to understand the game world” on both a narrative and gameplay level (2011); Fernández-Vara describes a “moment of illumination” (2009, p. 123) when the

⁸First described in (Reed et al., 2014a) with the slightly less self-explanatory term “eurekon.”

player understands the action necessary to “restore a pre-set behavior” (put the story world back on the track of the ideal walkthrough). In the context of an adventure game, this moment of awakening, understanding, and illumination is not just a matter of lateral thinking: it connects two worlds, the ludic plane of objects and actions and the narrative plane of the ongoing story. In a burst of insight, the player understands what story action will have the ludic consequence they desire and allow their traversal of the game to continue. The moment is unique in adventure games—at least when done well—because the insight is not just a logical solution but a revelation about the story, too.

All three pillars of the adventure game are necessary to enable and support eureka stories. Puzzles form the challenge to be overcome, a roadblock in the path of a traversal that force the player to reconsider their understanding of the narrative and the world. Exploration and experimentation give the player a mechanism to challenge that understanding and collect more information, a technique not always available in pure logic puzzles. The story may have provided explicit or implicit hints before the puzzle solution is found, and provides the motivation and reward for solving it. Bringing all three of these elements into balance is what enables a eureka story; when they do not support each other, puzzles become unfair, unfun, or broken.

How do designers craft a eureka story? While the core of the experience is that momentary flash of insight, the full narrative dynamic actually requires a set of carefully balanced steps to produce the designed build-up and payoff. Specifically, the player must go through all the following stages in order for this dynamic to realize its full potential:



Figure 2.3: Scenes from *King's Quest IV*.

1. be unsure what ludic action will advance the narrative
2. explore, experiment, and consider possible actions
3. have a moment of revelation connecting the narrative and ludic planes
4. prepare and carry out a plan
5. feel validation and satisfaction when the plan succeeds and the story continues

For example, in *King's Quest IV* (1988) the player character, Princess Rosella, must at one point befriend a unicorn to continue her journey. The creature is shy, and bolts whenever you try to approach, which leaves many players stuck (1). Elsewhere in the world (2), however, the player can find Cupid bathing in a pool, having left his bow sitting nearby. Approaching him scares him off; he leaves his bow behind. What the designers intend is that the player will suddenly realize (3) that Cupid's bow is not a weapon but a way to make a creature fall in love. Returning eagerly to the unicorn (4), the player shoots the unicorn with the bow, and is satisfied (5) to see this work: the unicorn now dotes on you and the story can continue.

Of course, as many adventure game players can attest, this sequence does not

always play out as the designer intended. We can productively analyze many of the failures of adventure games as situations where these steps have not occurred in the order intended, because of bad luck or poor design: where the eureka story failed.

- If the player’s solution is different from the designer’s, (5) fails: the player feels let down or angry that their clever idea doesn’t work.
- If the player has an idea but can’t figure out the right input to enact it (the correct words in a text game or the right spot to click in a graphical game), both (4) and (5) are interrupted.
- If a player can’t find the solution through exploration and experimentation, they miss out on (3) through (5) and the rest of the story.
- If the stuck player becomes so frustrated by a puzzle that they look up the solution in a guide, or have it spoiled for them, they bypass (3) and (5), missing out on much of the dynamic’s pleasure.
- If the player accidentally stumbles across a solution without understanding it, through random luck or brute force, they also bypass (3), (4) and (5) and thus much of the intended satisfaction.
- Finally, a game that makes the solution too obvious bypasses (1), (2), and often much of (3): the moment of revelation is less satisfying if it comes too easily.

We can see that rather than being a simple ineffable flash, the eureka story is actually a quite complex narrative dynamic that requires careful design work to set up and pay off. I have previously placed the failure of this pattern as the root cause of the downfall of adventure games (2014a) though this idea is hardly new to me (Wolpaw 2000; Fernández-Vara 2009, p. 135). From a design perspective, we can see that this is a serious problem: some of these fail cases reduce the player’s enjoyment, but in others, they actually prevent the player from being able to continue playing the game.

Since eureka stories involve all three of the adventure game’s core pillars, it follows that understanding how those pillars connect is critical to crafting good eureka stories. We can best discover their interconnections, and how they can support or subvert each other, by considering them as corners of a triangle.

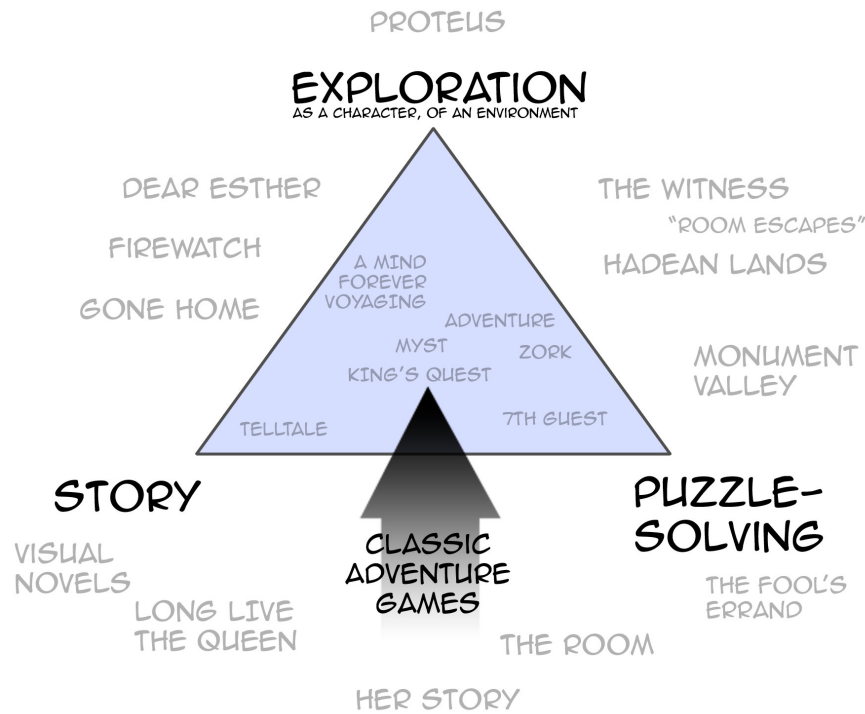


Figure 2.4: The Adventure Triangle, with adventure games inside it and adventure-adjacent games clustering around related pillars.

2.3 The Adventure Triangle and its Flaws

Let’s conclude our formal analysis of adventure games by considering the three pillars of adventure games as a unified structure: a triangle which encloses the space of classic adventure games, and around which we might imagine conceptually similar games

clustering, depending on which pillar or two they most strongly connect to (Figure 2.4). We might make several interesting observations about this triangle.

Each edge connecting two pillars can be thought of as a tension between two extremes. Each edge of the triangle connects two pillars, and can be seen as a tension between seemingly contradictory tendencies (Figure 2.5). The edge between story and exploration, for instance, can be characterized as the tension between a desire for narrative coherence (the story end) and player agency (the exploration end). The Story-Puzzles edge is a tension between flow (a story, naturally wanting to move forward) and challenge (a puzzle, putting up an obstacle to that progression). Finally, the Puzzle-Exploration edge represents a tension between solutions (the single correct answer to a puzzle) and possibilities (of open-ended exploration).

An immediate corollary to this is that each point of the triangle is in tension with the two others. Puzzles interfere with narrative progress and impede player exploration, for instance. A classic adventure game attempts to simultaneously resolve all three of these tensions to create eureka stories, but as we will see, this is made difficult by the close interconnectedness of the pillars.

The tension in each edge is both a promise and a challenge. Consider that a particular game might resolve the tension in each edge either successfully or unsuccessfully, creating either a positive or negative dynamic during play. If the tension is resolved successfully, it has been turned into a strength: a productive synergy between two seemingly opposing points. On the other hand, a failure to resolve the tension produces

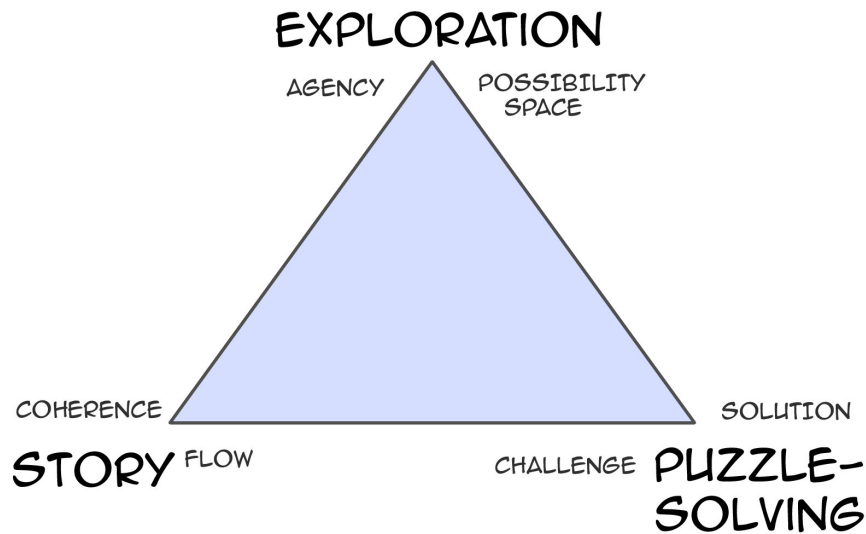


Figure 2.5: Edges on the adventure triangle.

frustration or dissatisfaction with two competing urges: at worst, the edge might “snap,” breaking one or both of the connecting pillars.

These three positive and three negative outcomes are illustrated in Figure 2.6. For example, successfully balancing the tension between coherence (story) and agency (exploration) can produce a dynamic where the player feels like the star of the story. The player character’s narrative is progressing, and the player feels as if this is a result of their exploration: the story is responsive to, not limiting, their actions. On the other hand, failing to resolve this tension can break one or both pillars. Too much player freedom might break a scripted story event if the player gets the game into an unforeseen state or kills a major character who still had plot points to reveal; too much story can break the player’s sense of exploration, and make them feel railroaded into a narrative they can’t participate in. Likewise, failing to resolve the Story-Puzzle tension means the

story stalls out because forward progress is impossible; successfully resolving it increases the player's understanding of how story and puzzles connect and their sensation of a coherent, well-crafted universe. Finally, failing to resolve the Exploration-Puzzle tension means the player feels dumb because the results of their exploration and experimentation aren't producing the required solution; but successfully resolving this tension gives the player a clever sense of having figured something out.

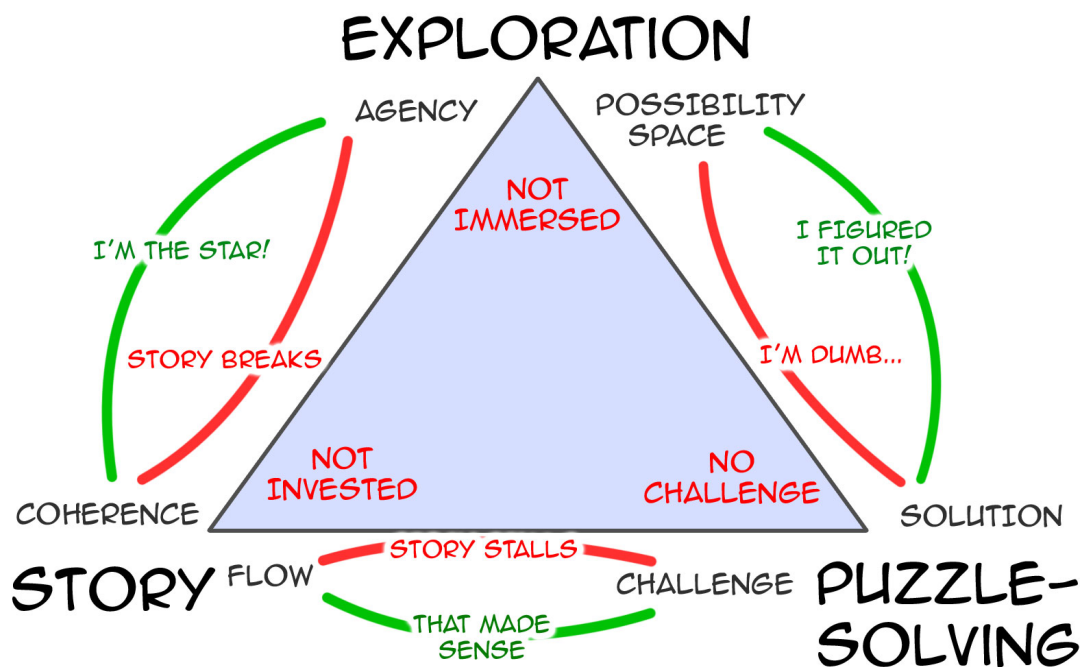


Figure 2.6: Promises and challenges on the adventure triangle.

Removing one pillar will strengthen the promise of the opposite edge and weaken the dangers of the two connected edges. Each pillar (point) on the triangle has an opposite edge (connecting the other two points). Removing a pillar should make it easier to produce the opposite edge's positive effect, and remove the

possibility of the failures of the two connected edges (since the point they connect to no longer exists).

For example, if a designer wanted to strengthen the player's sensation of starring in a story, they might try removing puzzles. This prevents the issue of the story stalling out (since there are no longer challenges the player must overcome to advance it) as well as the issue of the player feeling stupid (because there's no longer a hidden solution that must be uncovered). The designer can now focus on the aspects of exploration and story that enhance each other, increasing the chances that they can successfully produce the positive dynamic they're aiming for. This is what's done in *Firewatch* (discussed below in 2.5.2).

However, removing a pillar also removes the promise of its connected edges, which introduces a related new weakness. As a consequence of the same designer removing puzzles from their game, it would be more difficult to achieve the positive dynamics of the player feeling like they'd figured something out (with no solutions to find) or for the player to gain a holistic understanding of the story by working to build its next plot point (by encountering a successful eureka story with a narratively situated puzzle). Losing both these pleasures might result in the game feeling too easy, without enough of a challenge to make the rewards feel worthwhile. Similarly, removing exploration lessens the sensations of starring in a story and applying learned knowledge to solving problems, and introduces the additional problem of not feeling immersed in a world; and removing story removes the possibility of understanding a narrative through

enactment, as well as any pleasures from roleplaying the protagonist, while raising the risk that the player will not feel invested in the outcome.

2.3.1 Instability

We can view the decline of adventure games as stemming from the inherent difficulty in balancing all the tensions of the adventure triangle. We might also see it as symptomatic of copying the surface-level mechanics and conventions of adventure games without a deep understanding of the underlying structural reasons these components existed. The 1999 “Is Adventure Dead?” GDC panel that foregrounded story in the genre’s definition also put some of the blame for its decline at the feet of shallow copycats, which copied the conventions of bestsellers without the same level of deep design thinking: “every publisher wanted a *Myst*-killer, which spawned a huge crop of ‘me-too’ games that mostly sucked” (Lowe, 1999). Puzzles often became complex for their own sake, or to arbitrarily inflate play time, losing the connection to a sensible world that could allow a player to understand and solve them (Gilbert, 2004). A famous takedown of a complicated series of ridiculous actions in *Gabriel Knight III* concludes “Who killed Adventure Games? I think it should be pretty clear at this point that Adventure Games committed suicide” (Wolpaw, 2000). Puzzles with obscure or tortured solutions do not reward players for exploring the environment, instead making them spend most of their play time in a state of frustration. Conversely, games that took away too much of the player’s agency in the hopes of making puzzles easier deadened the core sense of immersion and freedom that many found charming about the genre in the first place.

While classic adventure games continue to be produced today, many are made as nostalgia pieces without much design innovation.⁹ The genre in its original form is no longer represented in bestseller lists and at awards shows, and nor does it enjoy the same cultural cachet it had amongst early gamers. However, fondness for its three pillars—and in particular, attempts to recapture the pleasures of the eureka story in different guises—still guide the design thinking behind much storytelling in games, and there have been many attempts to reinvent or reclaim elements of the genre in different forms. Some designers are trying to recapture the spirit of the genre by reinventing it in different ways.¹⁰ We will shortly look at three specific examples of games trying to reclaim parts of the adventure game magic, with three very different solutions.

2.4 *Blue Lacuna*

Before looking at modern reinventions of the genre, I'd like to briefly digress to discuss my own contribution to the space, *Blue Lacuna* (2009).¹¹ A hobby project that eventually got rather out of control,¹² I have previously written (2010; 2011b) about the game's goals and innovations.¹³ Created explicitly as an homage to adventure games

⁹Some recent examples include *Broken Age* (2014; from the creators of *Monkey Island*), *Obduction* (2016; from the creators of *Myst*), a remake of *King's Quest* (2015), and the Wadjet Eye *Blackwell* series (2006-14; inspired by early '90s graphical adventures).

¹⁰In addition to the game discussed below, other examples include *Life is Strange* (2015), *The Wolf Among Us* (2013) and other games from Telltale, *The Vanishing of Ethan Carter* (2015), and *Gone Home* (2013).

¹¹The game can be found at <http://blue-lacuna.textories.com>.

¹²At the time of its release, *Lacuna* was most likely the largest text adventure ever created, with some 375,000 words of prose and Inform 7 source code. Other games have since reclaimed this title with lengths approaching or exceeding a million words, including *SLAMMED!* (2014) and *80 Days* (2014).

¹³Others have also written about the game, as my dissertation committee encourages me to point out: including Parakenings (2010); Short (2011); and Ensslin (2014, p. 105-122).

(and also well before any of the design thinking in this chapter had taken place), it's worth briefly discussing how the strengths and limitations of the genre as described above played out in my own earnest attempts to master it.

Seen from a high level, my goals to make a great adventure game were largely realized through trying to make each of the pillars bigger and better. Players could explore a huge, dynamically changing island, with a day/night cycle and weather that continuously altered the text. The story had designs to be Serious and Important, and centered around interaction with an extremely complex NPC, Progue (who alone represented nearly a third the code and prose in the game) who was highly responsive to the player's interactions. I also took seriously the challenge of integrating the puzzles into the fictional environment. For example, Progue, a painter, has built a locked door preventing access to a part of the island that contains traumatic memories for him, and which he doesn't want outsiders to see. The lock shows a series of colors and a set of sliders with different icons representing plants and animals. The player is meant to experience a eureka story where they realize they've seen that shade of lemon-yellow before, in the leaves of one of the trees on the island—and come to understand that Progue's paints have all come from local materials. This is the key that lets them decipher the lock and get through the door.

Lacuna was widely praised in the interactive fiction world for its scope and ambition.¹⁴ As an adventure game, however, (and with the hindsight of a carefully

¹⁴The game won several best-of-year awards from text game makers, and has even appeared in all-time top ten lists of the medium (<http://ifdb.tads.org/viewgame?id=ez2mcyx4zi98qlkh>). I even suspect that *The Witness*, discussed below, contains an homage to it: <https://twitter.com/aaronareed/status/71742626823314308>.

constructed design framework) the approach of just making each individual element bigger predictably resulted in even bigger problems integrating the three pillars. The expansive exploration gave the player so much freedom that the elaborate mechanisms to advance the complex, dynamic story would frequently break: even in the final released version, after much bug fixing, it's still possible to find bugs derailing the ongoing narrative and thus putting the game in an unfinishable state. Like many of my adventure game designing predecessors, I also got too caught up in crafting elaborate puzzles (despite a special optional "story mode" that would remove some of their complexity) with the result that many players got stuck and never reached the intricate endgame sequences with their narrative payoff. Despite my attempts to narratively ground the puzzles, some of them ended up not being especially good eureka stories—including one I particularly regret now that involves learning the language of bee dances to find a secret in a forest.¹⁵ There's not much in the story to help players understand how to solve this puzzle, with the result that many never did.

But some of *Lacuna*'s success certainly comes from the places I did take care to closely couple the puzzles and the story, and some sequences I do remain proud of. Near the story's conclusion, you visit two different worlds which are each attempting to gain your trust and support. The player visits each world through a form of mental projection, inhabiting the body of one of its inhabitants. One world involves a seeming forest utopia: upon arrival, you borrow the body of one of many gardeners who wander the woods tending sacred trees with spades. You are given a welcome and a special

¹⁵It seemed like a good idea at the time.

headband to mark you as the Visitor, and then are free to explore—though with the strong sensation that an eye is being kept on you, and your experience is somewhat stage managed. What is never made explicit by the game is that if you remove your headband and grab an unclaimed spade, your impression of the world may be quite different. Much like a mystery shopper, you’ve gained the ability to wander the world without anyone knowing you’re special—and as a result can gain perhaps a truer, less polished understanding of how the society really functions, warts and all. Though this is not required to progress, and perhaps few players discovered this option, I’m pleased in the way it relies on understanding the narrative world well enough to realize you have a ludic affordance—a disguise—that can allow you to change your understanding of the story.

2.5 Three Design Solutions

While adventure games as a distinctive genre are no longer commercially popular, many strands of their design continue to be rewoven into new games today. In many cases, we can see these descendants as attempting to recapture something of the magic of eureka stories, a sensation largely absent in contemporary bestselling games centered more around following instructions than revelatory moments.

Specifically, I will consider three successful games released in the last few years, each of which attempts to resolve the dilemma of the adventure triangle by removing one of its pillars. *The Witness*, the highly anticipated second game from *Braid* developer

Johnathan Blow, removes the story pillar to focus on the pleasures of exploration and puzzle-solving. *Firewatch*, which has sold over a million copies, instead sacrifices puzzles, to focus on story and exploration. Finally, *Her Story*, winner of multiple Game of the Year awards, sacrifices exploration to strengthen the sympathetic dynamics between story and puzzles. Through a close reading of each game’s design decisions, I will demonstrate how a deep genre-specific theory of design can help to understand seemingly very different games in a unified context.

2.5.1 *The Witness*

In *The Witness* (2016), the player explores a *Myst*-like island filled with beautiful scenery and strange devices. Where *Witness* diverges from *Myst* most prominently, however, is that it deliberately and relentlessly removes all traces of story from the experience. It attempts to reinvent the adventure game by sacrificing story to strengthen the pleasures of exploration and puzzle solving.

Puzzles are the only mechanic. Puzzles in *The Witness* (some six hundred of them) are all variations on a single theme: drawing lines between two points on a grid, with various rules and combinations of rules defining what makes a correct connection. This is close to being the only mechanic in the game: the only two others are movement and activating audio logs (discussed further below). While successfully solving some puzzles causes a state change in the world (such as a door swinging open, allowing access to a new area), these results are all triggered via manipulating the puzzles themselves. The

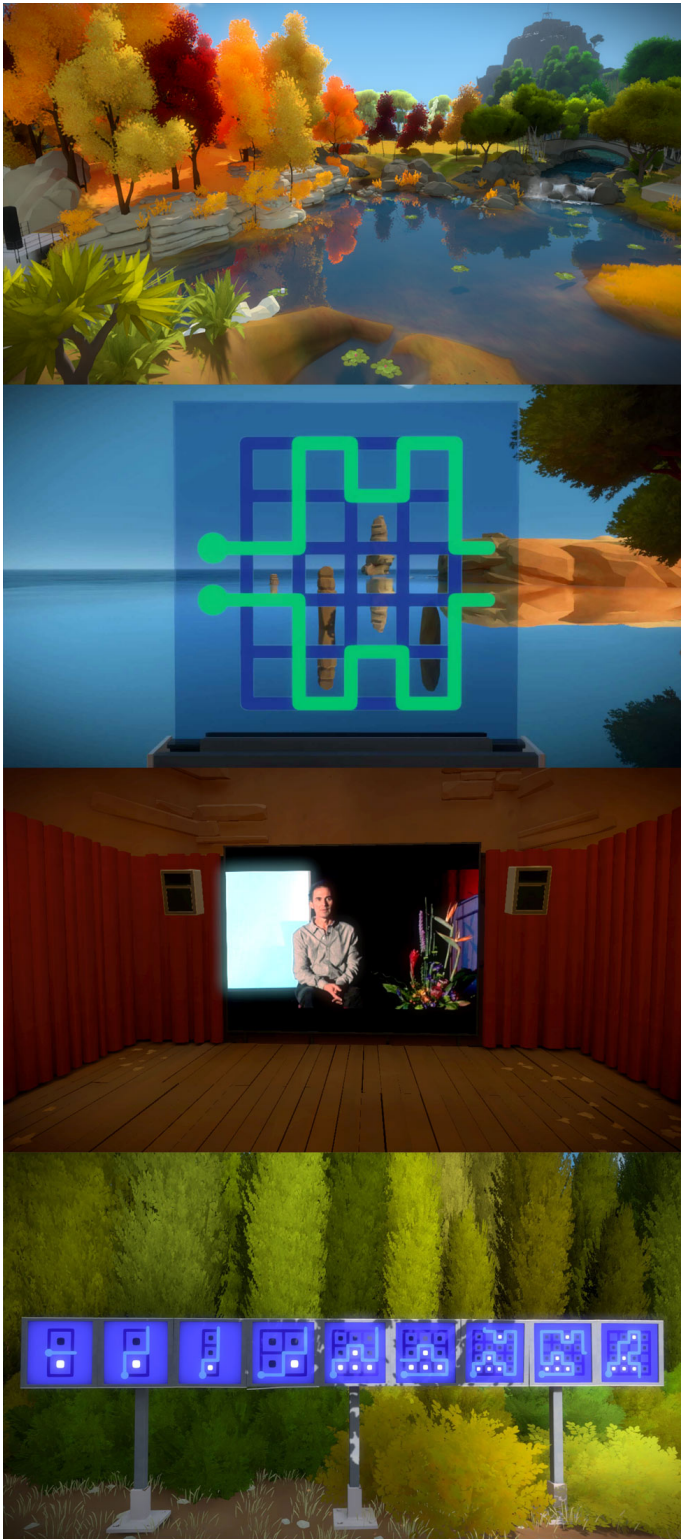


Figure 2.7: Screenshots from *The Witness*, which sacrifices Story to focus on Puzzles and Exploration.

player interacts with the world almost entirely through puzzle surfaces.

Puzzles are clustered around the island in themed groups. Each group centers around a new rule framework for how lines must be drawn (such as moving through or avoiding certain points, making a specific shape, enclosing particular symbols, and so on). Within each group, puzzles must generally be solved sequentially and proceed from simple to more complex challenges based on permutations of that rule or combinations with previously learned rules. To complete each group, however, understanding the rule is not enough. Most groups eventually or immediately require the player to also notice and integrate environmental cues into their solutions.

The environment is essential. *The Witness* features a meticulously crafted environment, and as the player progresses through the game they realize every detail is carefully constructed to enable a complete traversal. Other than doors opening and other changes to allow access to new areas, the environment is entirely static: the sun and clouds do not move, tree branches do not sway in the wind, and there are no characters or even animals present in the environment. This seems at first just a stylistic nod to early games like *Myst* that had to present a world as a static slide show due to technical limitations, but on closer examination this unchangingness is an essential part of the game.

As the player continues to explore, they will start encountering puzzles that cannot be solved without noticing details of the environment and how it interacts with puzzle surfaces. In one area near the game's starting point, the puzzle group cannot

be completed until the player notices that distant objects seen behind the transparent puzzle panel are marking the correct path. Other puzzles involve noticing the shadows cast by tree branches on the puzzle board, its reflection in a body of water, or the way the sunlight refracts off its surface. By the game's conclusion, it seems as if most if not all of the technical features of the 3-D engine—reflection, specularity, transparency, positional sound, colored and filtered light, the sound of footsteps, and so on—exist to provide solutions to one of the puzzle groups.

It's especially important to note that these environmental cues do not merely hint at an answer, in the way a narrative clue in a traditional adventure game might. They universally and unambiguously reveal the correct solution, once the player fully understands how this puzzle group relates to the environment, and has found the right perspective from which to view the puzzle. The world does not merely exist to support the puzzles: it does so perfectly.¹⁶ This idea, almost a *reductio ad absurdum* distillation of a core element of adventure games (that environmental observation supports puzzle solving), works by entirely excising another core element: story.

There is—emphatically—no story. *The Witness* has a remarkable and deliberate absence of story: the removal of narrative is nearly surgical in its extent and precision, and is worth breaking into its component pieces.

First, the nature of the player character is deliberately ambiguous, in a way analyzed extensively by Andrew Plotkin (2016), frustrating attempts to pin down even

¹⁶Momentary weaknesses of your frail human brain aside.

whether it is an abstract or specific entity. While the first-person perspective suggests the ambiguous protagonist of *Myst*, the player casts a deliberately gendered and costumed shadow. While the manipulation of puzzle surfaces seems to imply an invisible hand drawing lines across their grids, the puzzles can be touched from any distance, including at times from miles away. Since there are no objects to interact with in the world, this leads to a surprising conclusion:

Touch has nothing to do with it. You never manipulate the physical world (if there is one!) in any way. Indeed, if you look closer, the island is most unwilling to react to your physical presence. You can hear your footsteps, but you leave no footprints, nor even ripple the surface of a puddle you step in. You cannot brush aside a twig or pick up a bit of paper to read. You are a ghost, or a shadow of a ghost. Do you interact by observation? Perhaps you are simply recognizing the paths, and the panels react to that recognition.

The Witness has a player, but it's less clear whether it has a player character. It follows, then, that all the rewards it offers are purely based on the exploration and observation of the *player*—access to new areas, and the ability to perceive the (often beautifully rendered) objects within them.

That environment also goes to significant lengths to stay untainted by narrative. Many adventure games make extensive use of environmental storytelling: a player exploring a place with buildings and contraptions naturally wonders “who made this, how long ago, and why?” This instinct is so ingrained it takes some effort to overcome it in the case of *The Witness*, which upon closer examination avoids providing even implied answers to these questions. Many of the game's locations seem to exist solely to justify the others. There's a factory where the ubiquitous puzzle signs appear to have been manufactured; a greenhouse to grow the island's colored flowers; a quarry and

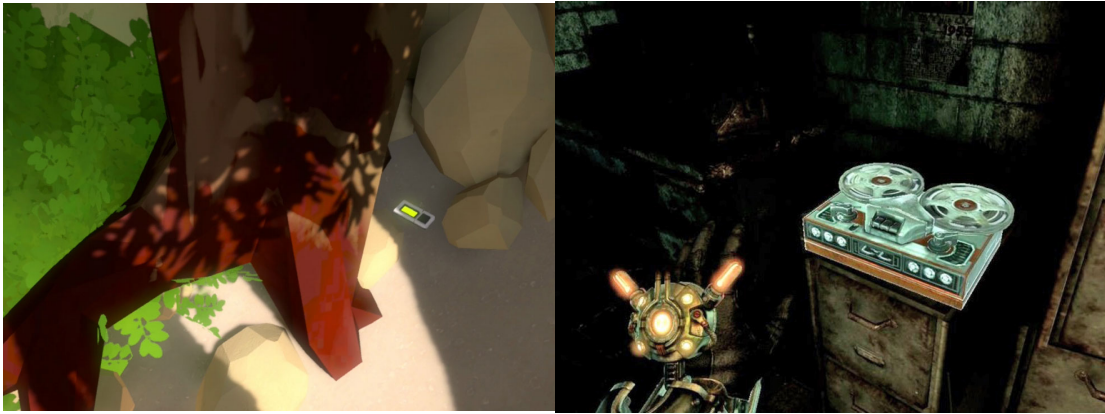


Figure 2.8: The audio logs from *The Witness* (left) deliberately evoke the artificiality of the standard “story dump” narrative mechanic in games like *Singularity* (2010, right).

sawmill to justify the marble and wooden buildings that contain these works. Much of the remaining architecture seems to echo the great pursuits of humanity: temples, tombs, and art studios (where humans have traditionally pondered questions like “Who made this, how long ago, and why?”) as well as fortresses, factories, and tree houses (symbolic of some of the classic distractions from those questions).

The game’s third mechanic (besides movement and puzzle-solving) is activating audio logs, stylized as digital media players scattered around the world: a clear evocation of the static, non-interactive way story lexia are presented in many contemporary games. Rather than containing recordings of characters who may have existed in this world or taken part in story events, however, each recording is explicitly extra-diegetic: a quotation, complete with spoken attribution, from a non-fictional person. As with the architecture, it’s clear these materials are not meant to be interpreted as creations of characters within the simulated world. The quotes are from philosophers, historians, astronauts, scientists, religious leaders, writers, and artists, and tend to involve contemplation of

purpose and the meaning of life: we might read them collectively as a meta-level version of the question we'd hoped activating them might answer, namely: "Why are we here? What's going on?"

The game also features a smaller number of video messages that can be uncovered and which are also explicitly extra-diegetic, with film credits or a vintage production company logo to make it clear they were not produced for the game. One of these, a long clip (excerpted in the quote below) of historian James Burke from the 1978 BBC television series *Connections*, speaks perhaps to the designer's project of so completely removing story from his game:

... Today, the people who make things change, the people who have that knowledge, are the scientists and the technologists, who are the true driving force of humanity.

And before you say, "what about the Beethovens and the Michelangelos?" Let me suggest something with which you may disagree violently: that at best, the products of human emotion, art, philosophy, politics, music, literature, are interpretations of the world, that tell you more about the guy who's talking, than about the world he's talking about. Second hand views of the world, made third hand by your interpretation of them. ... [Art] is easier to take, isn't it? Understandable. Got people in it. ... [S]cientific knowledge is hard to take, because it removes the reassuring crutches of opinion, ideology, and leaves only what is demonstrably true about the world.

The Witness wants to make the case that story in adventure games is a distraction. An enterprise based on the player synthesizing an understanding of narrative and ludic planes is doomed to failure, because interpretation of narrative—like all art—is inherently subjective. To solve a puzzle with a logical, definitive solution, one should only make use of what is logically, definitively—*demonstrably*, in Burke's words—true about the world.

In summary, *The Witness* abandons the story pillar of the adventure triangle to focus on maximizing the pleasurable dynamics connected to exploration and puzzle-solving. If we refer back to Figure 2.6 again, we can see that our theory predicts this should have the following positive effects:

- The player’s freedom to explore should no longer threaten to disrupt the story
- The puzzles should no longer be in danger of stalling out the progress of a story
- The dynamic between Exploration and Puzzle solving, of the player feeling clever for “figuring thing out,” should be enhanced

And indeed, we can make the claim that these statements accurately describe *The Witness*. Many game reviewers found the puzzles difficult, but fair¹⁷ and described the pleasures of exploring the game’s beautiful environments. IGN’s reviewer described the sense of flow as “freeing,” and summarized the strengths of the exploration-puzzle link succinctly: “It hooked me in with its masterful puzzle design and gorgeous visuals... a freedom granted by a world as welcomingly open to exploration as it is enjoyably challenging to solve” (Rad, 2016).

However, our theory predicts that removing story should have negative effects, as well:

¹⁷IGN used almost this exact phrase in their review (Rad, 2016). Kotaku’s review said “The more you figure out... the tougher and more satisfying *The Witness* becomes” (Totilo, 2016). GameSpot’s reviewer (Mahardy, 2016) has perhaps the most eloquent statement of this point:

The Witness is one of the most challenging games I’ve ever played. ... But when you do persist, frustration gives way to gratification. This is what makes *The Witness* special. Unlike many puzzle games, it doesn’t just make you feel intelligent—it begins on the assumption that you are intelligent. It trusts that you are a perceptive human being, capable of patience and critical thinking, and it rewards you for using both.

- The sensation of starring in a story should be reduced
- The player should understand less how their actions add up to a sensible narrative
- The player should feel less invested and motivated

Again, we can make the case that these statements are true of *The Witness*.

We’ve already quoted Plotkin’s analysis of the unsettling disconnect between player and player character in the game. Many reviewers also negatively commented on the game’s lack of story, such as this review in *Wired*:

The Witness is a sterile, lifeless videogame. It revels in the idea of knowledge, fascinated by how it’s earned and what it signifies. But it seems uninterested in players and their accomplishments, and with that lack of interest comes a lack of the human touch necessary to make sense of the knowledge it offers.

The Witness is like the island on which it takes place: a machine, to the core. Anything within it that seems lifelike is superficial. (Muncy, 2016)

The Witness trades the immersion and satisfaction of story to enhance the connection between exploration and puzzles. One can find the satisfaction of resolving problems via pure logic, and a kind of echo of eureka stories in the connection of environmental observation to puzzle solutions, but no narrative to place that sensation within. The next game we’ll look at makes a different trade-off, with different results.

2.5.2 *Firewatch*

Another acclaimed 2016 game that evokes the adventure game aesthetic is *Firewatch*. In the game, the player takes the role of Henry, a new park service employee looking to escape a rough situation, as he begins a job at a fire tower deep in a Wyoming wilderness. Through a fixed story arc spanning Henry’s summer on the job, the player

explores the surrounding environment and builds a connection with his supervisor Delilah, who exists only as a voice on Henry’s radio. The two build up a friendship while investigating a series of mysterious threats and learning the tragic history of some of the valley’s former inhabitants.

Firewatch tells a strong story and foregrounds exploration in a richly detailed environment, removing puzzles to keep the player immersed and the story moving forward. Of the three adventure pillars, this is by far the most commonly removed in modern storygames evoking adventure game aesthetics. After the success of games like *Dear Esther* (2008) and *Gone Home* (2013), some gamers coined the derogatory term “walking simulators” to describe these games with neither puzzles nor combat: a term which has been cheerfully reclaimed by game designers focusing on storytelling and exploration.

Structure. *Firewatch* features exploration of a single map which becomes more open as the story advances.¹⁸ As the story progresses through a series of episodes (corresponding to intertitles marking how many days Henry has been on the job), narrative events cause new areas to become available, generally one at a time: gates become unlocked and new paths are marked on the map. While the player has a certain amount of freedom in their exploration (such as choosing which route to take to get to a distant point), there is rarely an actual choice to be made other than proceeding to the next area indicated. While the player is free to explore parts of the map not connected to the current episode, the story will generally not advance until they travel to the next indicated point. The player

¹⁸While this is similar to how many classic adventure games are structured, the designers were apparently more inspired by the classic Nintendo game *Metroid*, in which the player can freely explore a map, slowly unlocking new abilities or keys that allow them to reach more and more areas.



Figure 2.9: Screenshots from *Firewatch*, which sacrifices Puzzles to focus on Story and Exploration.

has slightly more freedom in their interactions with Delilah through a conversation-tree menu,¹⁹ but the overall narrative trajectory cannot be altered: and critically, despite playing on genre expectations about optional romances or multiple endings, Henry’s summer always ends the same way.

Mechanics. In contrast to the stripped-down simplicity of *The Witness*, the player’s space of actions and inputs in *Firewatch* is larger and messier. There are a number of significant mechanics:

- Many varieties of movement: walking, running, jumping over obstacles, climbing up and down ledges, rappelling up and down ropes
- Picking up portable objects, examining them from different angles, and choosing to keep or discard them
- Interacting with fixed objects (opening and closing cache boxes and fences, pushing over a tree, etc.)
- Manipulating various navigation tools (map, compass, tracking device)
- Conversing with Delilah via timed conversation tree; silence can sometimes be significant
- “Reporting” an object seen in the environment to Delilah, triggering either a static audio lexia or a conversation tree

These many mechanics help the player feel like the star of *Firewatch*: by embodying the many actions Henry performs (fumbling with a map, jumping over logs,

¹⁹The developers have said there is “far more dialogue in the game than you could ever see in a single playthrough” (Chris Remo, Steam forum post, Jan 11 2016) and reviewers have noticed different conversations and story points appearing based on conversation choices. While admitting Delila’s story does not branch, the developers have called this customization a way of creating a “shared history” with Delilah that feels unique for each player (Palumbo, 2016). We’ll discuss this concept more in chapter 3 on sculptural fiction.

chatting with Delilah, hiking through the wilderness) the connection between story and exploration is strengthened. The wide variety of mechanics and objects/locations to use them on also allows the player to pick and choose which to interact with, which helps create a sense of ownership and agency, and lets the player focus their investigations on narrative lexia and plot threads that interest them. Contrast this with exploration in *The Witness*, where environmental clues cannot be missed, since each is vital to understanding the correct solution to a puzzle. Exploration exists in each game, but serves very different dynamics.

The environment too in *Firewatch* is more varied and dynamic, changing with the time of day and weather, reacting to the actions of Henry and off-screen characters. Trees sway in the breeze, fires start and spread, vandals wreck a previously orderly scene. In contrast to the static situation that must be apprehended in *The Witness* (supporting abstract puzzles with no temporality), *Firewatch* wants to appear like a dynamic world that's being lived in (supporting an ongoing story with forward momentum).

No puzzles. *Firewatch* takes pains to prevent the player from getting stuck, either in their exploration of the world or their ability to advance the story. The next objective is always shown as a banner along the top of the screen, and usually involves either performing an action on a nearby object, exploring a small area looking for something, or navigating to a specific area. Even navigation provides little challenge: the player has a map with an extra-diegetic highlight of their current location, and a compass.

Notably for our purposes here, the game has no puzzles that require the player to

have moments of revelation to uncover the next correct course of action. In a pre-release interview, designer Nels Anderson was asked if the game would include puzzles:

There aren't Myst-esque puzzles or anything like that. You'll certainly have to be engaged with the game's systems to make your way through the story, but you won't be needing to rearrange scrambled letters to spell a word or match coloured tiles or anything like that. (qtd. in Palumbo, 2016)

It's interesting to note that the puzzles cited as examples here are explicitly extradiegetic puzzles—arbitrary challenges that have nothing to do with the world or story in which they're situated. Rather than make a distinction between good and bad puzzles, and attempt to make more of the former (as *The Witness* does), another strategy is to discard puzzles entirely. This strategy makes perfect sense in light of the designers' project to make a game foregrounding exploration, a story with forward momentum, and the ongoing development of a close relationship between a strongly characterized player character and NPC.

In summary, *Firewatch* removes all puzzle-solving to strengthen the pleasurable dynamics associated with story and exploration. Again returning to the adventure triangle, we can expect this move to have the following positive effects:

- The flow of the story is no longer threatened by the tendency of puzzles to stall it out
- The player will be less likely to feel dumb for failing to observe how things in the environment are useful
- The dynamic between Story and Exploration, of feeling like the star of a story, is enhanced

Again, this feels like an accurate summary of the points covered above. Most

players will be able to uncover the full story of the game in only a handful of hours of play²⁰ without having to worry about coming up with correct solutions. The sense of identity with the player character is strong and enhanced by many other design decisions (the first thing the player sees on beginning the game, for instance, is the text “You are Henry”). Reviewers of *Firewatch* have frequently commented on the game’s immersion²¹ and the sense of connection with its two main characters.

Again, we should also expect to see some negative effects from the removal of one of the adventure game pillars:

- The sense of mastery/figuring things out should be reduced
- The player should feel less personally invested in the story because they did not need to come to an understanding of it to advance
- The player should feel less challenged without obstacles to overcome

Defenders of walking simulators have talked about the different game experience they can create: “It doesn’t matter if you understand it or it doesn’t matter if you ‘get’ it. It’s not a problem to be solved, it’s just a thing to be in for a while” (Dan Pinchbeck, qtd. in Campbell, 2016). However, detractors say removing challenges also removes engagement:

Walking forward is just a matter of pressing down a key or stick. And unless you are my dad playing a game, this doesn’t pose any sort of challenge at all. Your brain is basically unoccupied and the chance of your mind starting to drift is very high. Instead of being immersed in the game’s world you might

²⁰Contrast this with *The Witness*: as of May 2017, fifteen months after its release, less than ten percent of players had completed the central group of puzzles: <https://www.trueachievements.com/a221845/challenge-achievement>.

²¹*Game Informer*’s reviewer, for instance, describes being “immediately drawn into the game’s world” (Cork, 2016).

start thinking of what to cook for dinner or something else that is totally unrelated to the experience the game wants you to have. (Grip, 2017)

While one might take issue with the notion that one’s “brain is basically unoccupied” without gameplay challenges (such as while reading a book? watching a movie?) we might still observe that without explicit challenges, eureka stories cannot happen. We might have narrative revelations while exploring the worlds of *Dear Esther* or *Firewatch*, but they are disconnected from the ludic plane. They run the risk of feeling like they happened to the star of the story, not necessarily to (or because of) us. We did not choose for these things to happen to Henry—we did not make them happen—and we thus might feel less responsibility over the outcome.

These trade-offs, of course, are often made with deliberate intent. Much as *The Witness* incorporates lack of compelling narrative and interesting characters into its design and aesthetic, so does *Firewatch* intentionally integrate frustration of player choice into its interpretation. In a piece entitled “Firewatch took away our ability to be good people, and that’s where it shines,” reviewer Olivia White (2016) argues that the player’s lack of agency over the story—their inability to take game actions to affect it—plays into the understanding of both Henry and Delilah. White notes that Henry’s choices in the game’s prologue for dealing with his wife’s declining health are all selfish. Similarly, while certain dialogue options seem to suggest the player can influence Delilah’s decisions one way or another, her actions, including the pivotal decision she makes at the game’s climax, are always the same. Delilah is not a compliant character able to be manipulated by the player in the way most game characters are. The design limitation of removing

the player’s control over the story serves to make Delilah a more static, independent, and perhaps more realistic character.

2.5.3 *Her Story*

The third leg that might be removed from the adventure triangle is Exploration, but this has been tried far less frequently. Indeed, since on-screen avatars and navigation through spaces are such common game tropes, it might be hard coming from a blank slate to even imagine a story-based game without exploration. *Her Story*, an independent game that won a number of Game of the Year awards in 2015, takes an imaginative leap into an original game design space. While at first glance it does not look similar to classic adventure games, by analyzing it we reveal that it can be productively examined as yet another way to rehabilitate some of the genre’s lost pleasures by removing one of its core foundations.

In the game, the player operates a 1990s-era police computer with a database of video clips, taken from a series of interviews with a (fictional) woman accused of murdering her husband. The clips, each a response to a single question, are indexed by their transcript: a search interface lets the player pull up a list of clips where the woman says (for instance) the word “murder.” This mechanic drives the entire game: the player must work their way through the library of clips and attempt to come to an understanding of the woman, the murder, and a personal judgment of whether she’s guilty or innocent.

However, two carefully crafted limitations turn the experience of accessing the

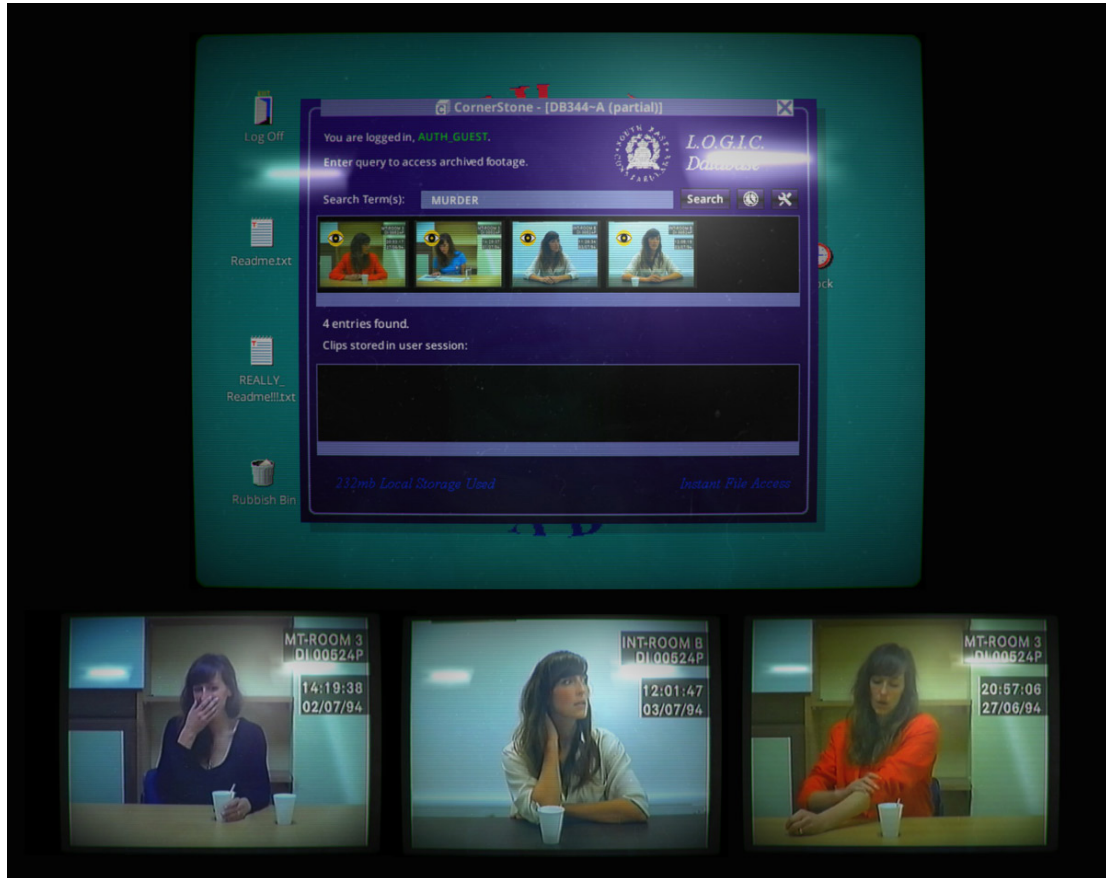


Figure 2.10: Screenshots from *Her Story*, which sacrifices Exploration to focus on Puzzles and Story.

clips into a game: (1) the player is given no index to the clips, and thus must come up with search terms by hearing them used in other clips or thinking of them on their own, and (2) the search engine will only return the first five matching results, meaning the player must come up with more specific words to access later clips. These limitations mean the player can only have a successful traversal (“solving” mystery) by coming to a greater and greater understanding of the story, through gaining enough knowledge to know what search terms will uncover subsequent clues.

A reinvention of expressive text input. One of *Her Story*’s most successful elements is its expressive textual input, an all-but-forgotten device in contemporary games. As in a text adventure parser, the clip database offers the possibility of typing in anything the player can think of, rather than choosing candidates off a list or picking from a prescribed series of actions. The cleverness of the game’s conceit is that the database metaphor makes the translation from input to action (to revisit our framework in 1.2.2) completely transparent. The player might not understand why a command in a text game like `ACQUIRE LAMP` did not succeed: perhaps the verb “acquire” wasn’t implemented, or the game doesn’t have a mechanic for taking objects; perhaps either or both words were misspelled, by the player or the designer; perhaps the lamp is accidentally or intentionally simulated as being immobile, or out of reach. By contrast, a word entered into *Her Story*’s database will either succeed (if the woman²² ever speaks that word in one of the clips) or fail (if she doesn’t) and the reason for each is obvious. The operational logic of

²²For reasons which become clear when one plays the game, I will avoid mentioning her name in this discussion.

searching for clips is entirely transparent to the player, which significantly reduces the frustration caused by the opacity of many text input systems.

Her Story manages to make both success and failure fun with this expressive input. Productive topics are in no way highlighted by the game: the player must notice when the character mentions a new person or subject that might be fruitful to investigate, and try typing it in on their own. The player is also free to try words sparked by thematic connections, sudden insights, or purely at random. The woman mentioning her sister might prompt the player to try searching for *brother* or *family*. My own first playthrough of the game included a delightful moment of satisfaction when I noticed that she drinks coffee in some clips and tea in others: searching for these two words revealed a significant clue. Another similar moment came not from an external hint but an internal revelation: realizing that in a story about questions of self-identity, the word *mirror* might be important.

This clever design means failing is also productive. Failure allows you to rule out certain subjects of inquiry: if a search for *father* produced no results, meaning the subject never once mentions it in any interview clip, the player can assume such a character is probably not part of the story. Failure also might produces many false positives (i.e., clips with the word *father* incidentally mentioned): the player must watch these clips to know whether they are significant or not, and in so doing inevitably encounters new lexia mentioning other topics, sparking new ideas for searches. The interface means the player can't help but drive the story forward as they navigate through more and more of the maze of footage, each spoken word connecting rhizomatically to all the other

clips that use it. In a well-designed game, “Failure serves the deeper function of making players readjust their perception. . . In effect, failure adds content by making the player see new nuances in a game” (Juul, 2008). But this kind of productive failure is often rare in storygames, which tend to have more constrained plots and thus many more ways to fail than to succeed (Juul, 2005, p. 73).

No exploration. *Her Story* removes the classic adventure game pillar of exploration. Some might quibble with this claim, given that the central mechanic is finding one’s way through a maze of connected clips. To defend it, we must clarify slightly what we mean by *exploration*—and it’s fascinating that it’s taken a game with such an original structure to force this.

In *Her Story* the player has no avatar,²³ and there is no simulation of physical space. While the database of interconnected clips might be read as a metaphorical space, crucially the player character does not occupy a specific point within it. They are free to jump to any node at any time, provided they know (or can think of) a word it includes. There is also nothing within that space to experiment upon: clips can be watched, but cannot be manipulated, and will reveal no further information upon closer examination (unless that information is details overlooked on a first viewing).

Another way of putting this is that the space of *Her Story* is both static and open: there is no way to alter its state, and there is no way to hide anything in it other than through the structure itself. All “links” (outgoing words) are immediately

²³The game at first does not seem to feature a defined player character, either: but in a complete traversal, the player character’s identity does in fact become significant.

transparent upon watching the clip; even unlinked clips can be accessed provided the player thinks of the right word to try. Nothing is locked, except through comprehension of the plot. *Her Story* features no exploration, in the sense we mean when discussing adventure games—of a simulated space, by an avatar situated within it—while retaining the genre’s other features of story and puzzle-solving.

Puzzles? We might also need to justify our claim that *Her Story* has puzzles, given that, as we’ve just discussed, the game is stateless. Not only that, it does not seem to have traditional puzzles at all, in the sense of specific roadblocks created by the designer and placed at a particular position in the world. Each player will encounter their own moments of puzzlement based on which plot threads they’ve discovered on their walk through the database of clips, and which elements of the story are obvious or unclear to them.

Again, we must view our definition of “puzzle” in the context of eureka stories and the adventure game framework to see that it does hold up here. Through this lens, a puzzle does not require us to change the state of the simulated world, nor is it required to be a specific, designed artifact.²⁴ It merely requires us to have a moment of revelation connecting the narrative and ludic planes—a moment where the player realizes what ludic action (typing a word) will result in advancing the story (accessing a new clip). This is *Her Story*’s standout dynamic: and as with *The Witness* and *Firewatch*,

²⁴It is not quite accurate to say *Her Story* does not have designed puzzles. The game’s creator has spoken about how the scripts for each clip were meticulously edited before the video shoot to avoid prematurely mentioning keywords associated with deeper parts of the mystery (Barlow, 2016). However, this still suggests “pruning” a space of emergent puzzles rather than designing puzzles individually.

it achieves this focus by sacrificing the elements of the adventure triangle that do not support it.

In conclusion, our theory states that removing Exploration should have the following positive results:

- The player should be less likely to break the story through exercising their agency
- The player should be less likely to feel dumb through a failure to solve challenges
- The positive dynamic between story and puzzle-solving, of feeling like your actions explain and justify a sensible story, is strengthened.

The structure of the story and the careful way clips are interconnected means the player has complete freedom over how they navigate. Though players might be stumped for a while if they can't think of new search terms, the “productive failures” enabled by the design make random guessing or trying words seen in prior clips very likely to suggest new avenues of investigation. Finally, reviewers have commented on how *Her Story* makes you feel like a detective: the direct link between solving the “puzzles” and advancing the story enhances this sensation that player is directly driving the narrative forward.

Again, we should also expect to see some weaknesses from removing Exploration:

- The sense of being the star of the story should be reduced...
- ...as should the satisfaction at figuring out the right puzzle solution through observing the environment
- The player should feel less immersed without an environment to explore

Our analytical lenses give us a deeper appreciation for how *Her Story* manages to feel so innovative and enjoyable when we consider how design decisions mitigate each of these weaknesses. While the player does feel less like the hero of the story, immersed within it—explicitly watching a character, rather than identifying with or controlling her—this ties into that character’s secretive nature, which is revealed to be an elaborate ploy around the nature of her identity. The mystery would not work if we began the game from her point of view, knowing what she knows. Likewise, while we do not get to enjoy the riddle-like satisfaction of solving well-crafted puzzles, the dynamics arising from navigating the tightly interconnected web of clips both create bespoke puzzles for each player’s unique traversal, and help provide solutions to them by continuously offering new suggestions of other keywords and clips to try.

While *Her Story* was obviously not designed with our framework in mind, we can see how one might design a game by starting with a study of the various trade-offs associated with discarding or exaggerating certain parts of the adventure triangle, and arrive at novel design ideas or inspirations. The structure allows us to predict in advance what the effects of these fundamental design shifts might be. By using three successful released games to verify our theoretical framework, we can show its usefulness and point the way towards creating further such useful frameworks for other styles of storygame.

2.6 A Technological Fix?

Each of the games examined above attempted to recapture some of the pleasant dynamics of adventure games through the design step of removing one of their core pillars. However, we might imagine another approach: easing the strain between the various competing needs through creating new technology.

The field of drama management, for example, can be seen as an attempt to resolve the tension between story and exploration by adapting the story to the player's actions (Weyhrauch, 1997). Applying algorithms for predictive planning and efficient state search, drama managers hope to bolster the sense of starring in a story by minimizing its connected weakness (breaking a brittle story through too much player freedom), rather than removing or minimizing the dynamics that enable it. Similarly, technological solutions have been proposed that we might read as strengthening the connection between exploration and puzzles, such as dynamic or adaptive systems for hints (Mehm et al., 2013) or difficulty (Hunicke, 2005); or for strengthening the bond between puzzles and story, such as through adapting puzzles to the narrative content the player has encountered (Fernández-Vara and Thomson, 2012). However, these solutions have often been hampered by the difficulty of developing or standardizing such technology, or the failure to integrate it into a practice of game production.

As storygame-adjacent technology continues to advance, however, we should not forget this second promising avenue for reclaiming the pleasures of eureka stories. In the 1980s, many game studios offered 1-900 numbers you could call to get recorded hints to

puzzles you were stuck on—though some players may have been more inclined to ask their game playing friends if anyone had figured that one out yet. New technological solutions may allow this process of *repair*—understanding why a storygame is not working for a particular player, and correcting it—to become integrated into the game itself, turning the stumbling blocks of the triangle from a vicious to a virtuous circle.

This lengthy look back at a well-established storygame genre has borne useful and varied fruit. Through a historical analysis of both contemporary and retrospective definitions for adventure games, we have teased out the three core pillars at the genre’s heart: puzzles, story, and exploration. This ground-up definition lets us consider this type of game through its core narrative logics, rather than as a factor of its interface (text parser or mouse clicks), presentation style (prose or graphics), or point of view (first, second, or third person). Through this framework, we can talk about a broader swath of games than are generally discussed or studied together through their shared narrative design patterns (Figure 2.11).

With this framework in hand, we can characterize and discuss complex design patterns arising from this style of game’s foundational elements. For example, I identified the “eureka story” as a design pattern wherein the game creators intend the player to experience a specific sequence of events centered around a moment of illumination where understanding of the narrative and ludic planes of the story comes into a new alignment. The framework provides tools for pinpointing why the eureka story is such a difficult design pattern to realize, and trace these to the fundamental difficulty of satisfying all



Figure 2.11: A ground-up definition of adventure games lets us discuss a similar eureka story—realizing the player needs to push over a dead tree to cross a gap—across genres not always studied together (clockwise from top left: *Limbo* (2010), a puzzle-platformer; *Space Quest II* (1987), a graphical adventure; and *Trinity* (1986), interactive fiction.)

three core pillars of adventure games.

Finally, the framework can point us towards various solutions to this problem, reinterpreting a commercially abandoned genre in a new light. For instance, we can predict that certain design moves around weakening one of these pillars to strengthen the others should have particular effects on players, and have seen these theories confirmed through three contemporary games that each weaken one pillar (*The Witness*, *Firewatch*, and *Her Story*). In addition, technical solutions such as drama management point towards other ways of resolving the genre’s design tensions, by suggesting new narrative logics that designers in the ’80s and ’90s were unaware of or did not have access to, and which in turn suggest new paths to bringing back aspects of the adventure game’s unique pleasures.

In the chapters ahead, I will apply a similar process to three emerging storygame genres: carefully constructing a definition, considering how games created by myself and others have begun to explore this design space, and turning theory into practical insights for designers.

Chapter 3

Sculptural Fiction

These are novels of potentiality. Quantum narratives. Their power isn't in their final acts, but in the profusion of superpositions before them, the could-bes, what-ifs and never-knows. Until that final chapter, each of those is as real and true as all the others, jostling realities all dreamed up by the crime, none trapped in vulgar facticity.

From "China Miéville On Crime Novels" (2009)

It is early 2013, and I am playing Yoon Ha Lee's text game *Winterstrike*, built with the StoryNexus engine. In this game, you play a survivor of the winterstrike, an at-first vaguely specified calamity that's struck the city of Iria, part of a far-reaching galactic empire. The winterstrike has brought on a deep, charmed winter, and also cut the planet off from all interstellar travel. Both your character and the city are struggling to remake their identities in a new world that's colder and stranger than it was before. The writing is beautiful and evocative, and draws me into the world. StoryNexus games are designed to only be played a limited number of moves each day: this has prevented me from getting sucked in to too many of them, but I find myself returning to *Winterstrike* day by day. I too have become trapped there.

Like all StoryNexus games, as you progress through the story you gain and lose various *qualities*. These sometimes represent familiar numeric elements in games, like money, points, or faction alignments. *Winterstrike*, however, has an intriguing quality called *Ice* which is not explained. As I take various actions in the world, I try to figure out what Ice represents. It seems to increase when I take cruel actions, so perhaps it means “cold-heartedness?” And yet at times I am unable to take an especially bold or daring action because my Ice is not high enough. Other actions become limited because my Ice is *too* high. I realize the concept Ice represents—at least in my interpretation—is slightly more complex.

As I approach the end of the story, I realize that I’m going to have to make a decision about what kind of person my character is. There are actions I can take that will increase my Ice, and open up an endgame path I’m particularly interested in. But I’m not sure my character is willing to do them. The logic that makes that pathway available is based purely on the numeric value of this score—the specific actions you take to change qualities are generally not remembered by the system, only their numeric impact—but *I’ll* remember how I got that Ice. I realize that *Winterstrike*, using the affordances of StoryNexus, is asking me to make a moral choice in a way that has a subtle but powerful difference from those in games that give it to me in a list of three options. I will need to *build* my own story of how I became the person who had enough Ice to reach this particular goal, out of the various opportunities in the story world that allow me to increase Ice. Perhaps if I keep investigating, I’ll find better ways to increase this stat—better, for my own personal sense of my character and their morality. But



Figure 3.1: Screenshot of *Winterstrike*.

perhaps my character has grown weary of looking already. I spend a few days wandering aimlessly through the same old locations and actions, trapped like my character, unsure if we are ready to become a person who can move on.

The first new storygame framework I'd like to discuss I call **sculptural fiction**, a mode of interactive narrative that centers building a story graph, rather than traversing an existing one. *Winterstrike* and StoryNexus only partially support this vision—much of the mechanics are about choosing from a small set of options, rather than expressively acting—but they point the way towards a style of game that becomes more reflective and more creative. Developed originally as part of my MFA projects and thesis (2011b), the idea of sculptural fiction has evolved through the construction and release of several major game projects, which I will describe below and in chapter 4. Sculptural fiction's contribution to game design can be positioned as a pragmatic approach to a particular problem in the design of storygames: I will lay out this problem and how a new mode of storygaming can help solve it below.

In this chapter, I will first define sculptural fiction and how it differs from more established narrative logics. I will connect it to earlier modes and aesthetics, including hypertext theory, sculptural construction in the artistic and therapeutic senses, and the pre-digital origins of literature with sculptural components. I will consider several existing systems that implement ideas similar to sculptural fiction, but find that none of them entirely capture the core of the aesthetic as defined here, and are thus unable to fully realize its potential. I will discuss two prototypical projects of my own as I worked to develop an aesthetic of sculptural fiction (*18 Cadence* and *Perfect*). Finally, I will discuss

the challenges with authoring sculptural fiction and how these can be addressed through both design thinking and technical approaches. This complete theory of sculptural fiction is the primary design contribution of this chapter. My technical contribution is focused in the subsequent case study chapter on *The Ice-Bound Concordance*, where I will discuss how a working sculptural fiction engine was developed and integrated into a successful independent game.

3.1 Sculptural fiction overview

Sculptural fiction developed as a reaction against the limitations of stories based on choice graphs (discussed in 1.2.7) and how much this narrative logic has come to dominate story-based gaming. Recall that in this framework, the player navigates a fixed story graph where edges are presented as choices between possible courses of action. The player occupies a specific node on the graph corresponding to the player character's position in the narrative.

At their worst, a player of a game driven by choice graphs can feel like a rat in a maze:

- Because most such games do not allow the player to reverse a decision, and because the player cannot generally see the structure of the graph, **decisions are high-cost and stressful**. Both ludic and narrative consequences of a choice may be unclear, meaning a choice may lead the story in a direction the player finds unsatisfying.
- Designers struggle to create graphs complex enough to simulate meaningful choice, but simple enough to be fully authored and error-free.

- Since a story graph is static, the player can never make a choice that surprises the designer: all possible pathways have been determined in advance.
- A small list of choices means the player does not generally have expressive input (see 1.1.3), and may not feel the sense of engagement and creativity this can allow.

Designers certainly can and have created compelling storygames driven by choice graphs,¹ by working within these limitations or turning them to their own advantage. But I have often found myself frustrated as a player by the disorientation and limited viewpoint of being embedded in a choice graph, and as a designer by the time spent authoring huge chunks of “maze” that many players might never see, in order to provide an illusion of agency. While the AAA storygame *Heavy Rain* (2010) was heavily hyped for its (choice-graph based) interactive narrative upon its release, its designer David Cage urged people to only play the game once, lest they “kill the magic of it” by revealing how little their choices actually mattered (qtd. in Gaskill, 2009). Magic tricks are fine;² but I wanted to explore a different space of storygame, where meaningful choices could

¹I have indeed created a major work in this mode myself: *Hollywood Visionary*, briefly discussed in (3.3.3).

²There’s an inherent difference, however, between illusion of agency and understanding of agency. The former traffics on a game player’s familiarity with existing narrative logics (such as an expectation that choice points will have certain kinds of effects), or the remarkable human ability to explain seemingly contradictory things (see my discussion of Storywriting in 7.2.4.2, and of *Life in the Garden* in 3.5). A game that allows and invites deeper consideration of its narrative logics, however—that lets one understand how they work, and provides something interesting to understand—offers the designer a new channel of expression. As the player through play and replay (Mateas, 2004, p. 27) comes to understand the narrative logics and ludic structure of a storygame, they gain a deeper appreciation of its narrative plane and how it connects to these expressive processes: of what the designer meant to convey through the union of the two. Wardrip-Fruin (who advances the term “expressive process” in his book of the same name) writes about the *Eliza* effect (2009, p. 25), after the original 1960s chatbot: when a system’s seemingly complex facade breaks down upon closer examination into a disappointing reveal of its simplistic internal nature, a “pay no attention to the man behind the curtain” moment for digital systems. Designers should work to either maintain the illusion—keeping the curtain firmly in place—or to ensure that multiple encounters with the wizard provide increasing and revealing insights into how he works.

be more than illusory.

Sculptural fiction attempts to address these shortcomings by giving the player a different kind of role. Rather than centering the act of navigation through a fixed graph of story nodes, the primary focus becomes the act of exploring the space of possible connections. Through mechanics for linking and unlinking nodes, the player builds a story, rather than moving through an existing structure.

This gives rise to play dynamics where many of the above concerns are addressed:

- Decisions are lower-cost and less stressful, because (1) they can be reversed, providing a space for exploratory consideration of possibilities; and (2) considering nodes themselves rather than simply the edges between them provides more information about the likely narrative consequences.
- The author shifts from building a static multicursal graph with many delicately interconnected nodes to building a smaller, more flexible set of nodes with rules for how they can be connected into unicursal stories.
- Assuming a large enough set of possible nodes, player input becomes iterative and expressive, opening the possibility of slowly improving a story towards a desired final state, of player creativity, and of surprising the designer.
- By centering exploration of a possibility space rather than choosing between fixed-length paths, the length of play time becomes more in control of the individual player.

The term “sculptural” is meant to suggest the constructive and iterative process of the artistic practice of sculpting, particularly the tradition of “assemblage” where existing objects are arranged to form an artistic result (Rawson, 1997, p. 15). The term also suggests the sculpture itself, an artifact produced by this act and designed to be exhibited and shared; and also the small, continuous decisions in viewing a finished

sculpture, such as what angle to view it from and for how long.

Sculptural fiction implies (but does not require)³ a higher-level perspective on a story, where rather than being embedded at a particular node the player can consider many possible nodes at once, and potentially add them to any part of an existing graph (corresponding to earlier or later moments in a narrative). The graph constructed by a player may be multicursal (building a story with multiple possible routes) but is more likely to be unicursal (building a linear sequence of nodes), since a multicursal structure, maze-like, is easier to process when embedded within it than when viewing its messy entirety all at once.

Expressive input strengthens the positive dynamics of sculptural fiction. As input becomes less expressive (that is, as the number of available nodes for the player to choose from becomes smaller, or the number of places they can attach them to shrinks) the narrative dynamics revert to those of simple selection, and the mode loses much of its power.

Having every lexia available at once, however, can be overwhelming, more akin to a jigsaw puzzle. Sculptural fiction is enhanced by narrative logics that guide the player towards narratively interesting nodes. These logics may consider the player's prior actions, the current state of the graph they've constructed, or the qualities of the unused lexia in determining what pieces to suggest. With no such architecture (meaning any node is available to add to the graph at any time), the player might be overwhelmed

³One could imagine a storygame where the player remained embedded at a particular node in a story graph, but progressed by choosing between visions of possible futures, adding available nodes onto the graph at their present position and then moving into them.

with possibilities, many of which might not be narratively satisfying. On the other hand, too restrictive of an architecture lowers the expressiveness of an input as described in the previous paragraph, leading to an experience that feels like choosing from preset options. The challenges of providing some curation without overly restricting player choice will be discussed below.

Sculptural fiction also implies a higher-level view of a story. In the choice graph logic, the player identifies with a player character at a particular node of the graph and a particular moment of a story. To effectively construct such a graph, especially when we might add or remove pieces from any point along it, the player is most likely considering the story from a less embedded perspective. This creates a tension between player engagement (in the narrative) and player understanding (of its structure) which will be discussed more thoroughly in (3.4.2.3).

To conclude, recall our definition of storygame:

- a playable system, with
- units of narrative (lexia), where
- the understanding of both, and the relationship between them, is required for a satisfying traversal.

Sculptural fiction is a specific mode within the space of storygames that provides these additional features:

- an expressive exploration of possible structures built from provided lexia;
- a system that highlights interesting lexia for the player to consider; and

- (usually) a high-level (rather than embedded) view of a story, to better understand its overall structures

Several of these qualities and features exist in earlier systems or are presaged by pre-digital experiments. Digital sculptural fictions, indeed, are built on foundations and key design questions that have roots in pre-digital practices. I will next consider some of these antecedents and how they have informed my thinking about sculptural fiction.

3.2 Connections to existing concepts and works

3.2.1 Pre-digital works

While a strong pre-digital thread foreshadows the rise of agency in stories, this is mingled with (and not quite the same as) sculptural fiction's primary concern of building stories from narrative pieces. For instance, various kinds of audience participation have been explored throughout the history of the theater, such as in the courtroom drama *Night of January 16th* (Rand, 1936). As the play opens, a law clerk steps in front of the curtain:

CLERK. Ladies and gentlemen, you are to be the jurors in this case. Twelve of you will be drawn to perform this duty. You will kindly set up here, take your seats, and receive your instructions from Judge Heath.

Ushers select audience volunteers, who do indeed sit on stage in the jury box through the entire performance and ultimately have the freedom to render the fictional defendant guilty or innocent. Multi-path stories such as Queneau's *Un Conte à votre*

façon (“A Story as You Like It”; 1967; reprinted in Motte, 1998) or *Choose Your Own Adventure* books (Packard, 1979) also let the audience choose between several pre-defined outcomes. This kind of decision, however, connects more closely to the history of choice graphs: we’re looking rather for artifacts where the player can genuinely contribute something creative.



Figure 3.2: Raymond Queneau’s “Cent mille milliards de poèmes” (Hundred Thousand Billion Poems), (1961)

Closer is something like Raymond Queneau’s “Cent mille milliards de poèmes” (Hundred Thousand Billion Poems; 1961), a sonnet of ten lines where each line has ten alternatives. The lines cannot be reordered, but even so this presents a tremendous possibility space, clearly far too large to enumerate as a list, and a reader might feel duly proud of a particular creation they make with Queneau’s literary contraption. The cut-up fiction of William Burroughs (1961) went even further down the constructivist path (though in this case only for the author, not the reader) relaxing any constraints

on positioning and embracing the trade-off of removing nearly all guiding structure to allow increased possibilities for serendipity.

The space between constraint and construction was explored in several experimental novels during the mid-twentieth century. In 1962, French writer Marc Saporta published *Composition No. 1*, a novel consisting of 150 loose and unnumbered pages, each with a few paragraphs of text describing an incident in the life of a man known as X. Accompanying instructions invite the reader to

...shuffle these pages as though they were a pack of cards. To cut them, if he wishes, with the left hand, as if he were consulting a fortune-teller. The order in which the leaves emerge from the game will dictate the destiny of X.

A few years later, English novelist B.S. Johnson unveiled a similar project, *The Unfortunates* (1969), with only slightly different instructions:

This novel has twenty-seven sections, temporarily held together by a removable wrapper. Apart from the first and last sections (which are marked as such) the other twenty-five sections are intended to be read in random order. If readers prefer not to accept the random order in which they receive the novel, then they may re-arrange the sections into any other random order before reading.

While Johnson's segments are longer, printed on pamphlets of as many as twelve pages, and his novel is given a slightly more specific ordering constraint (a mandated introduction and conclusion), the experience of reading both is much the same. Both stories are concerned with memory and death, suggesting a connection between the rhizomatic structure of remembrance with the randomness introduced by removing a pre-planned structure. Neither novel offers an obvious "definitive" version, nor does the experience seem much changed, for better or worse, by different orderings of pages.

A third fragment novel, Julio Cortázar's *Rayuela* (Hopscotch), first published in 1963, came conventionally bound, but with these instructions:

In its own way, this book consists of many books, but two books above all.

The first can be read in a normal fashion and it ends with Chapter 56 ... the reader may ignore what follows with a clean conscience.

The second should be read by beginning with Chapter 73 and then following the sequence indicated at the end of each chapter. ... Each chapter has its number at the top of every right-hand page to facilitate the search.

Unlike the Saporta and Johnson books, here the procedure is not random: it involves a conscious choice by the reader between two alternatives. Implicit in Cortázar's instructions is the notion of comparison between the given alternatives: either before choosing⁴ or after choosing, evaluating between the two versions and deciding which one is preferable.

The move towards conscious comparison and decision-making is an important evolution towards sculptural fiction ideas. In my reading of *The Unfortunates* I felt limited by the requirement of randomness: I began hunting through the stack, looking for other fragments connected to the one I was holding, trying to trace the parallel plots; laying related fragments on the table side by side to compare them; starting to read one fragment, then deciding to come back to it later and shuffling through to one that seemed more interesting in the moment. "[H]ow the mind arranges itself," Johnson's protagonist muses in one fragment, "tries to sort things into orders, is perturbed if things are not sorted, are not in the right order, nags away."

⁴Though this is difficult without knowing the content of the story, the reader might assume, for instance, that the second approach will produce a more fragmented, less linear narrative.

These projects also each offer different pre-digital approaches to the questions considered by sculptural fiction. What is the most productive size for a narrative fragment: a word, a page, a chapter? Does the reader need a frame to make sense of a story built from pieces, or other tools to assist in assembly?

A more subtle invitation to sculpt meaning comes in the form of works like Nabokov's *Pale Fire* (1962), which reveals how the construction of meaning can be aided or frustrated through literary devices and fictional framing. The book is presented as a long verse poem with a series of scholarly annotations. The narcissistic commentator, a second-rate academic named Charles Kinbote, does his best to overshadow the poem itself in his increasingly rambling and digressive endnotes, where he discusses not only his short-lived relationship with the deceased author but a seemingly irrelevant fascination with a far-off kingdom called Zembla and its exiled king. As the reader progresses through the work, the true nature and meaning of the contents becomes more and more uncertain. *Pale Fire* also comes with instructions for reading, presented during Kinbote's Foreword:

Although [the endnotes], in conformity with custom, come after the poem, the reader is advised to consult them first and then study the poem with their help, rereading them of course as he goes through its text, and perhaps, after having done with the poem, consulting them a third time so as to complete the picture. I find it wise in such cases as this to eliminate the bother of back-and-forth leafings by either cutting out and clipping together the pages with the text of the thing, or, even more simply, purchasing two copies of the same work which can then be placed in adjacent positions on a comfortable table. . . (p. 28)

There is, in fact, no straightforward way to read *Pale Fire*, despite its conventional binding. Even if one fails to follow Kinbote's elaborate instructions, the Foreword

references several endnotes, many of which immediately refer to other endnotes; the index includes recursive loops and mysterious hints; the central poem is long and difficult to appreciate without the context of the commentary, which is even longer and clearly from an unreliable narrator. The bothersome “back-and-forth leafings” seem in fact the only sensible way to proceed: flipping between poem and commentary and index with fingers holding prior pages, striking off on tangents and trying to remember the gist of multiple parallel threads of plot at once: attempting to assemble meaning from a series of deliberately scattered fragments.

“*Pale Fire* invites readers to discovery in a way no other novel does,” says Brian Boyd in his book on Nabokov’s novel: “from the very beginning [it] sets up a rhythm of successive discoveries” (1999, p. 1). Boyd quotes a contemporary critic, Mary McCarthy, who called the book “a Jack-in-the-box, a Fabergé gem, a clockwork toy, a chess problem, an infernal machine, a trap to catch reviewers, a cat-and-mouse game, a do-it-yourself kit.” Revealingly, nearly all of McCarthy’s analogies suggest puzzles or games, each one an elaborate construction.

Here is another aspect of sculptural fiction presaged by pre-digital works: the construction of meaning as an aspect of reading any literature, which can be foregrounded (or frustrated) through deliberate design. The advent of computers created a vastly more powerful platform for exploring these notions, and it is to sculptural fiction’s digital antecedents we now turn.

3.2.2 Hypertext Theory

The advent of digital literature opened new affordances for authors and readers, many of which have been explored by a community of hypertext theorists and practitioners over the past thirty-plus years. One of the key ways digital literature differs from its forebears is in how “the process of interpreting the text, trying to construct a story from the nodes available to us, is made more obvious to the reader” (Walker, 1999). The passive act of interpretation can be made a more active process through making texts ergodic and multicursal.

Many of the constricting aspects of choice graphs mentioned earlier have been analyzed in the context of hypertext theory. Landow (2009), for instance, writes about the disorientation problem in hypertexts: not knowing where you are in a network, how to get somewhere else, or understanding “the boundaries of the information space,” the range of possibilities. Some hypertext theorists including Landow have also positioned multicursal texts as potentially infinitely recombinable. Marie-Laure Ryan, however (2001a) has rejected the notion that “hypertext is like a construction kit: it throws lexia at the reader, one at a time, and tells her: make a story with this.” She argues that this argument is problematic:

Textual fragments are like the pieces of a jig-saw puzzle; some fit easily together, and some others do not because of their intrinsic shape, or narrative content. It is simply not possible to construct a coherent story out of every permutation of a set of textual fragments, because fragments are implicitly ordered by relations of logical presupposition, material causality and temporal sequence.

Ryan’s analysis is certainly true for many common structures in hypertext

fiction: for instance, the device of presenting a series of moments from an apparently linear story in a nonlinear order. A structure like this implies a underlying truth which the work's surface deliberately fractures and makes inaccessible. This suggests that for a work to be genuinely sculptural,⁵ there must be legitimately new stories that can be made from its source material: multiple, emergent ways to assemble the jigsaw.

While little existing hypertext fiction focuses on acts of sculptural creation by the player, the hypothetical system “Card Shark” (Bernstein and Greco, 2004) presages many of the ideas in sculptural fiction. Repurposing play mechanics from card games, Card Shark imagines a “deck” of lexia where each card contains rules for when its piece of narrative is allowed to be played. Rather than positioning the reader at a particular lexia to choose a specific outgoing link, the player of Card Shark can play any card that's currently valid according to its rules. Some plays may change the state, making some cards unavailable while opening up the chance to play others. This changes the dynamics of reading a hypertext in a fundamental way, by turning the primary mechanic of play away from navigating a graph and towards construction: forging a path through a dynamic, continuously changing possibility space.

While Card Shark was never directly implemented, the StorySpinner Sculptural Reader implemented some of its ideas to create “a test bed for experimenting with

⁵The term “sculptural hypertext” has been used previously, but in reference to an authoring practice rather than a mode of operation: beginning with all lexia linked together and intentionally removing links rather than additively joining nodes together (Bernstein et al., 2002). Sculptural authoring in the traditional additive sense—visualizing nodes in a graph as a structure that can be visually reconfigured—appears as an interface paradigm in tools like StorySpace and Twine, and has gone by the more formal label “spatial hypertext” (Marshall and Shipman III, 1995). Sculptural fiction as discussed here applies this paradigm (and acts of structuring, arranging, constructing and so on) to the reader of the text rather than the author, foregrounding the process of meaning-making that underpins much hypertext reading.

the authoring of narrative flow in automatically generated stories” (Hooper and Weal, 2005). Readers progressively selected tarot cards, which in turn showed lexia tagged with themes matching the card selected. StorySpinner, however, was never used in a completed and released story.

3.2.3 Other Theoretical Work

Other traditions have touched on notions of sculptural construction. The study of children’s block play, for instance, positions this activity as a productive middle ground between open play (such as free drawing, with little guidance) and closed play (directed activities such as worksheets with right and wrong approaches). The shapes and affordances of a set of blocks encourage certain kinds of activity (such as building towers) without restricting freedom so much that a child feels forced into a specific action, thus losing their intrinsic motivation (Wellhousen and Kieff, 2001).

Tabletop roleplaying theorists have also invented strategies similar to sculptural fiction in their efforts to give players more agency in a shared story space: this has been called “sandbox” play, rather than the “railroading” or “on rails” style where the gamemaster has a predetermined plot to move players through. Roleplaying games, however, are usually less about assembling existing material and more about generating new lexia improvisationally during play. We will discuss these games in much more depth in chapter 7 on Collaborative Storygames.

Another lens is provided by Erica M. Kleinman’s research (2016) into digital games that make use of “metagaming mechanics,” actions by the player that manipulate



Figure 3.3: Screenshot from *Save the Date*.

the reality of the fictional world rather than performing actions within it. In the context of narrative games, for instance, Kleinman identifies a set of games that offer a “rewind/redo” mechanic, where the player must replay an earlier sequence or restart the game entirely in order to proceed, and where the game acknowledges and responds to this action. Kleinman studies the small but intriguing body of games that have used metagaming mechanics to put the player in a different kind of role than simply embodying the protagonist, including *Life is Strange* (where the protagonist learns she can reverse time), *The Stanley Parable* (which explores the meaning of choice and interactivity in games through a set of variations on a single vignette), and *Save the Date* (a choice-based story in which your dinner date mysteriously dies no matter what choices you make; as you keep replaying, the protagonist becomes aware of these deaths and strives through knowledge of the choice structure to find a way to save her). Sculptural fiction offers another kind of metagaming mechanic, giving players affordances to directly

manipulate the story. The metagaming mechanics in each of the games Kleinman studies have a fundamentally different structure than those without such mechanics, and her work suggests authors must embrace a different aesthetic for interactive stories to truly make their multicursal nature foundational.

3.2.4 Released Games and Systems

Several released or publicly exhibited games and systems have used ideas similar to sculptural fiction, though none encompass all the qualities included in its definition here. Studying how these projects were implemented and considering their aesthetics as playable systems will begin to point us towards some of the strengths and weaknesses of sculptural fiction.

3.2.4.1 StoryNexus

The StoryNexus engine (Kennedy, 2012) from Failbetter Games implements many of the ideas in Bernstein’s Card Shark. Its most famous game is *Fallen London* (2009; originally known as *Echo Bazaar*), but the engine has driven other games released by Failbetter and by third-party authors using the StoryNexus platform (such as *Winterstrike*). StoryNexus incorporates several sculptural ideas into its design, which is worth describing in detail.

In a StoryNexus world, the player is defined through an unbounded set of numeric “qualities” that might represent attributes (such as strength or intelligence), goods (such as money or treasure), events in the player’s personal history, affiliations or

loyalties, or metatextual properties like score or narrative progress. Using the same card game metaphor as Card Shark, in StoryNexus a player's qualities determine what lexia cards (called "storylets") they may draw from a deck of opportunities. Each storylet describes a narrative situation the player encounters, and offers one or more possible responses. Responses may in turn be gated based on qualities, and may require a test against a quality to be successful, with the test becoming easier as the quality score rises. Each outcome of a storylet may alter one or more qualities, which in turn may change the available storylets that can be drawn from the deck.

For instance, in *Fallen London* a character may play a storylet where they encounter an "insufferable poseur" reading poetry in a bar, and choose one offered response: to mock him. This option tests the player's Persuasive quality: the outcome is random, but becomes more likely as this quality grows higher. On a success, the player's Persuasive rises, as does a Wine quality representing the player's stock of spirits (connecting to text mentioning others in the bar buying the player drinks). On a failure, the text instead narrates a humiliating rejoinder: the player's Persuasive also rises, indicating they are learning from their mistakes, but the quality Scandal rises as well. As a result of this action, the player's Persuasive may have risen to a point that new charisma-based storylets are available; they may have gained enough Wine that they can sell it to buy something more useful; the player's Scandal may have reached a point that a storylet related to being socially ostracized appears; and so on.

While the core framework is straightforward, StoryNexus offers a great deal of



Figure 3.4: Screenshot from *Fallen London*, showing several StoryNexus features, including qualities in the lower left, an active storylet in the center with several options, the last of which is tested on a quality, and the limited stock of actions in the upper left.

syntactic sugar⁶ over some of the core mechanics of sculptural fiction, and a system for providing authorial control over presentation and pacing. The system’s designers and writers have created a large number of conventions and high-level design patterns which make the system most effective, many of which have only emerged and evolved over time. For instance, *Fallen London* features a mid-level organizing structure, “ventures,” which consist of a linked chain of storylets: usually a beginning, a middle which must be performed multiple times to raise some temporary quality to a threshold value, and a higher-stakes ending tested against the temporary quality. Success often results in a significant material reward that might open up new storylets, ventures, or locations. The creators have codified at least sixty different design patterns (Arendt, 2010) for the interaction of qualities, storylets, and ventures, ranging from simple concepts such as barring a venture from appearing while the player has too much of a certain quality, to more complicated structures such as “Faust’s Tea Party,” where a player initiates an interaction with another player where both individuals will gain some amount of a desirable quality but lose some amount of another desirable quality.

StoryNexus demonstrates some of the potential of sculptural fiction (a large input space; a structure not based on choice graphs) although it was designed to produce a different kind of experience. While in larger StoryNexus games like *Fallen London* there can be enough potential storylets available at once for the input space to feel genuinely expressive, and the system’s gated opportunities are a mechanism for pointing the player towards the content most relevant to their current qualities, a sculptural

⁶A term of art in programming for a bit of code designed to make other bits of code more readable.

evolution of StoryNexus would include several key differences. First, playing a card in StoryNexus cannot be taken back and there are no save games, meaning choices remain consequential and potentially stressful. In addition, the player is still embedded at a particular position within the narrative, so cannot see their position with the overall structure or always see how to achieve a specific goal. These design choices increase immersion but limit the player’s ability to shape a story to their liking and understand how to find and unlock content they’d like to encounter.

3.2.4.2 *King of Dragon Pass*

An inspiration (Dunham, 2011) for *Fallen London* was the 1999 game *King of Dragon Pass*, in which the player manages the resources of a clan of people in a fantasy world and responds to events that threaten them or advance their story. Events can be gated based on the game’s state, and each choice then alters the state in different ways. As in StoryNexus, choices can succeed or fail based on stat tests.⁷ A scripting language allows for roles to be cast and text to vary based on complex nests of state-checking functions. For instance, a scene could narrate a rescue by the strongest neighboring clan with a positive attitude towards the player’s clan.

One key distinction between *Dragon Pass* and *Fallen London* is that in the former, the player receives a single event at a time, and must respond to it before moving on, with no ability to reject it or choose another. This makes it less sculptural than the newer game, where the player has dozens or even hundreds of possible storylets

⁷The system in *Dragon Pass* allows two stats to be tested against each other, an interesting wrinkle which StoryNexus does not support.



Figure 3.5: Screenshot from *King of Dragon Pass*

they can try next. While both games give the player an ability to build a strategy and identity through their choices, in *Dragon Pass* the player tends to react rather than act, responding to events thrust upon them by the system. As with *Fallen London*, players cannot take back moves or understand how choices will affect the larger plot, another way it differs from the ideal sculptural aesthetic.

3.2.4.3 *Storyteller*

While unreleased, Daniel Benmergui's prototype game *Storyteller*⁸ connects to a different aspect of sculptural fiction than the above two titles. In this game, the player is given a series of narrative challenges with a story condition defining victory, such as

⁸An early prototype was exhibited at and won the Independent Games Festival's 2012 Nuovo award, but as of this publication the final game had not been completed or released.

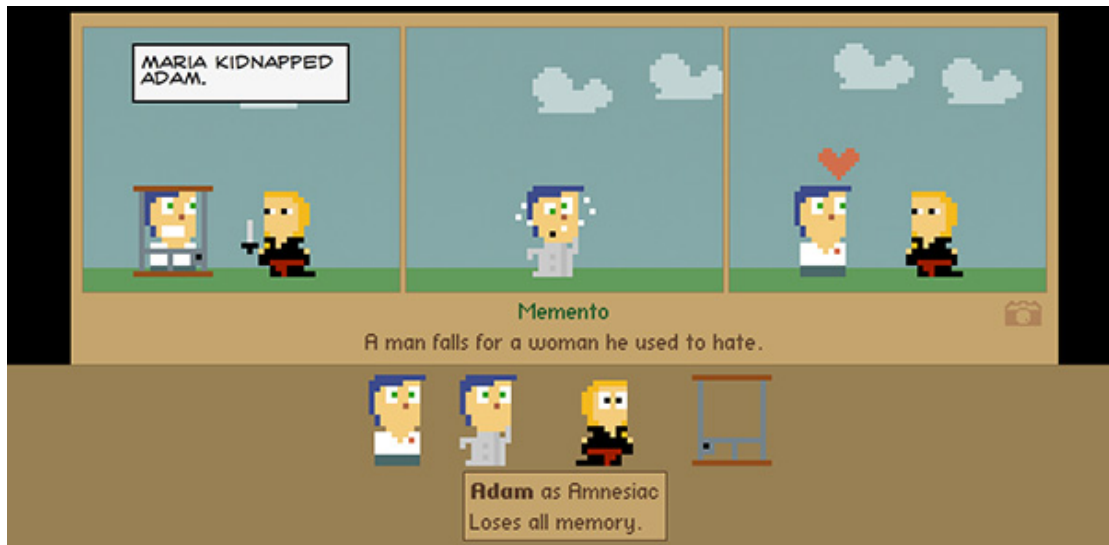


Figure 3.6: Screenshot from the unreleased *Storyteller*.

two characters getting married. The challenge is presented in the form of a multi-panel comic (Figure 3.6). Players can move characters and props in and out of the panels to change the state of the story world as of that moment in time. For instance, placing two compatible characters next to each other in the same panel might make them fall in love; putting one of them next to a tombstone in the next panel might make them make the other heartbroken. The stories do not have specific correct solutions like adventure game puzzles, however: there might be multiple or surprising ways for the player to solve one.

Storyteller shares sculptural fiction’s goal of low-cost, reversible decisions, and the ability to see the entire narrative at once: changes immediately propagate across the panels of the comic (visible all at once), encouraging a space of experimentation rather than embedded immersion. The set of possible story elements is visible on the screen in the form of characters or objects the player might place in a panel. *Storyteller*’s simple graphical stories are necessarily simpler than stories told with written lexia: this has the

advantage of player and system working on the same level of abstraction, but perhaps the disadvantage that the system’s stories risk being uninteresting by being “all bone” with none of the meaty, interesting details that make us read stories in the first place.

3.3 Early Sculptural Fiction Projects

While the games mentioned above explore aspects of my vision for an aesthetic of sculptural fiction, none capture all of its qualities at once. As my work developed I wanted to push the aesthetic further, reducing the cost to decisions by making them reversible and giving players a higher perspective on a story than that of a character embedded within it. I explored these ideas in the two games described next.⁹

3.3.1 *Perfect*

Perfect is a minimalist instantiation of the sculptural fiction aesthetic, giving the reader a set of sentences they can manipulate by moving differently colored squares on a screen.¹⁰ A set of five sentences shapes a memory of a perfect day (Figure 3.7). By moving the squares, the player alters certain aspects of these sentences along a spectrum of words that change their meaning in a discrete or linear way.

For instance, the second sentence describes the setting of the narrated event. For each sentence, the player has four axes (x and y for both squares) along which they can alter it. The meaning of the four axes is hand-authored for each sentence.

⁹These projects were both begun as part of my MFA thesis work and released after its completion. More details on the original prototypes of each can be found in (Reed, 2011b).

¹⁰Playable online at <http://aaronareed.net/perfect/>.

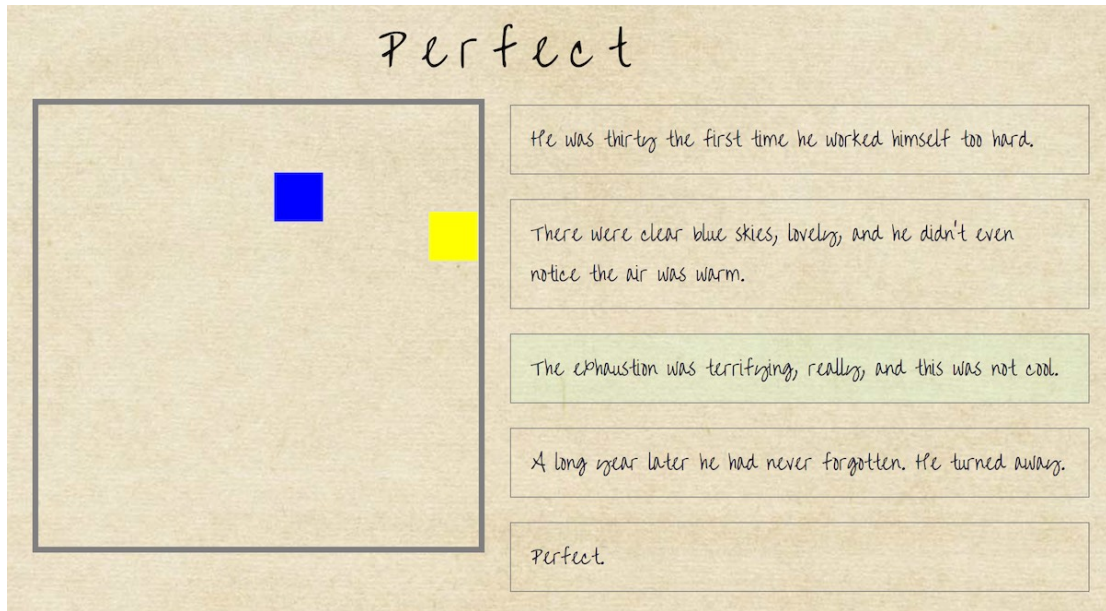


Figure 3.7: Screenshot from the author's minimalist sculptural fiction game *Perfect* (2011). The story seen here, written in 2017, describes the process of dissertation writing, a narrative possibility not considered when the original lexia were created.

In the second sentence, moving the yellow square left or right changes the weather from, at one extreme, “clear blue skies,” to, at the other, “wind and rain lashing the trees,” with incremental changes in between. Moving this square up or down changes the temperature; moving the blue square left or right changes the narrator’s perception of the event along a scale from “achingly beautiful” to “hideous”; and moving this square up or down changes how the narrator perceives these details, such as “not even noticing” them. Tapping a sentence changes the person of the narrator (between He, She, I, You, We, and They). Some sentences dynamically affect each other: for instance, changing the event in the first sentence from “kissed a boy” to “tasted a peach” also changes a reference in the third sentence from “his lips” to “the juice.”

In combination, these controls give the player the ability to significantly change

the individual sentences and the overall content of the story, in ways sometimes surprising to both the author and the player. Here are a few possible variations on the second sentence:

A magnificent rainstorm was raging, but the air was burning hot.

Only a drab wisp of cloud was in the sky, and we imagined the air was cold.

There were clear blue skies, marvelous, and you wished the air were frigid.

The goal was to provide enough affordances for players to feel a sensation of sculpting as they manipulate the story, mapping large movements to large changes and small movements to smaller changes, until they find an arrangement of squares that produces a sequence of sentences they find satisfactory. The original prototype for the piece called for making the metaphor more literal by instantiating the squares as blocks on a physical table, tracked via something like the Reactable platform (Jordà et al., 2005).¹¹ Even without this step, however, I felt the piece was a useful experiment in exploring the aesthetics of making the crafting a fiction feel tactile and sculptural, and connecting the player to a narrative even while remaining outside it, not taking the role of a character.

3.3.2 *18 Cadence*

My next project exploring a sculptural aesthetic positioned it within a more long-form story. *18 Cadence* imagines a fictional house that the player explores in both space and time (Figure 3.8). The lives and stories of its residents are told in

¹¹Wardrip-Fruin (2010) writes about the challenges of “complex surfaces” (repurposing a term from John Cayley) as interfaces to fictional structures, including not only works that involve physical objects but works where story content is presented in forms other than legible text.

single-sentence chunks anchored to a particular year and location in the house. Each year/location coordinate also contains a set of items that were present in the house at that place and time. Also shown are the name and age of each resident present there, each of whom has their own version of the narration which the player can toggle between.

Any of these pieces of text (events from a character's perspective, character names, ages, items, locations, and years) can be dragged into a canvas area below the main window and freely positioned. Fragments can be dragged off the canvas to be discarded. More interestingly, however, the pieces retain memory over what they are describing, and the system knows how to describe all the components in various combinations. This enables the user to combine multiple fragments from the same scene by dragging them together, or split them apart again with a razor tool. The system is able to render any combination of fragments as a complete sentence. If multiple possible renderings exist, the player can cycle them by tapping a fragment. Fragments are also positionally aware: putting two about the same character near each other, for instance, might change the character's name in the second fragment to a pronoun like "She." On a purely aesthetic level, fragments are angled slightly based on how near to the edge the user drags them, to create a feel of a messy analog assemblage (as with cut-out words or magnetic poetry), and fragments from earlier periods are styled as if the paper they are on has aged.

When the player has assembled, recombined, and positioned a set of fragments they like—making a story, according to their own definition—they can share it by uploading it to a server. This generates a link they can use to keep for future reference



Figure 3.8: Screenshot from the author's game *18 Cadence*.

or share with others. Previously uploaded stories can be browsed at random.

What makes an *18 Cadence* story? I deliberately left this vague. Certain goals are implied by the available material, such as reading all the content, creating and sharing a story, or using the workbench to discover and document the story of a single character, or connections between characters from similar or different time periods. Other goals may be created or discovered given the available affordances, and indeed one of the reasons behind the sharing mechanic was so these goals might be made visible. For example, nothing prevents creating multiple identical fragments, opening up the possibility of filling the entire workbench space with repetitions of a single word. The ability to alter the way the fragments are described suggests the possibility of attempting to unify the narration style, by cycling every fragment to (for instance) the shortest

available version. Fragments from different scenes do not join together, so they can be overlaid, allowing people to conceal or edit words from one fragment with the content of another. Empty workbench space can be used intentionally, such as by creating a story from a single fragment placed in the center of the page. And in fact, all of these variations and many I had not considered were attempted by various players: over four hundred stories were made upon the project's release.¹²

Cadence develops a sculptural aesthetic in several ways. Once the user orients themselves to the physical and temporal shape of the house, they can easily access all the raw material they have to work with: nothing is hidden behind links or special commands. The user is not embodied as a character within the story, and is indeed given no narrative role at all, even structurally. Changes are small and reversible: all operations can be undone, and nothing can permanently remove available options. (A particular story might be lost or destroyed, but it could always be recreated.) Finally, the user can spend as much or as little time as they wish navigating the text: there are no specific goals about when they have read enough content or created enough of a story.

Deliberately experimental, *Cadence* is arguably not a game, lacking features like a win state or challenges to overcome (Juul, 2005, p. 6). The “story” it tells is also deliberately vague and shapeless: the lives of the house’s residents tend to not resolve in dramatically satisfying ways. It’s up to the player to impose a narrative on these events through what they choose to include and leave out. From a narrative perspective, it moves farther in the direction of user control than most games do—players have a great

¹²I catalogued some of my favorites at <https://lacunagame.blogspot.com/2013/04/some-favorite-cadences.html>

deal of control over how they can access and manipulate the story lexia. However, the game enables this by moving away from attempting to understand the player's input and react to it, or simulate a world for the player to investigate. *Cadence*'s environment is a static tableau.

However, this was the first project that I felt truly realized all the facets of my concept for sculptural fiction, and upon release was noted by many¹³ as a bold and interesting step for games. Several reviewers picked up on the sculptural aspects and the way the game attempted to democratize the telling of an interactive story:

The player's involvement isn't a process like being an actor or a co-author, a director, a stage manager: you're an editor. Someone else has already shot all the footage, and it's your job to choose which material to use, and where. The cutting-board is far, far too small to hold everyone's story at once: if you use it to keep track of key events as you go through the full century, it'll quickly overflow. You have to decide whose stories, which elements of those stories, to focus on. (Ashwell, 2013)

The ideas explored in *Cadence* were followed up with an even more ambitious project, *The Ice-Bound Concordance*, described in depth in chapter 4.

3.3.3 *Hollywood Visionary*

I would also like to briefly discuss a game of mine that is not directly sculptural, but instead explores whether some aspects of a sculptural aesthetic can exist within a more limited structure, such as a choice graph. *Hollywood Visionary* is a 150,000 word choice-based fiction published by Choice of Games in 2015. Choice of Games releases

¹³*Cadence* was selected by Kirkus Reviews as a Best Book App of 2013, and received an Honorable Mention in the Nuovo category at the 2014 IGF Awards. It also received an Honorable Mention for the 2014 Robert Coover Award for a Work of Electronic Literature.

mostly traditional “choose your path” stories, with a particular house style based around the kinds of choices their readers find most interesting and compelling. *Visionary* is the story of a young movie mogul in the 1950s trying to produce a film that will launch a new studio, and I wanted to see if I could make this act of creation personal enough, even within a limited multiple choice format, that players felt creative ownership over their movie. In addition, I wanted to provide players with more elaborate tools for identity construction, rather than (for instance) choosing their gender off a list: again, with the goal of providing the players expressivity over the movie mogul character they were creating.

Visionary gives you a large number of opportunities to make decisions about the kind of movie you’re making, remembers them all, and uses them whenever possible in templated lexia (see Figure 3.9). A combinatorial genre system allows you to make, for instance, a “dark musical western” or a “religious science-fiction comedy.” Lots of minor decisions about the script, casting, technical choices, and priorities while editing affect a set of nine core variables about the film’s quality (in addition to the text labels and things like who is starring and what their role is). Similarly, you are given a number of opportunities to make statements about your character, including how you’d like employees or friends to address you, what you wear, who you’re interested in developing a (romantic or platonic) relationship with, and how you juggle priorities in your life.

Visionary is only minimally sculptural (in the aspects described above), and ChoiceScript’s limited affordances for varying text and storing state meant the game involved a lot of awkward coding and hard-to-catch bugs. But I did feel the experiment


```

The performance,
*if actingStyle = "improv"
    fleshed out with off-the-cuff improvisation,
*if actingStyle = "script"
    carefully sticking to the text of the screenplay,
*if actingStyle = "director"
    *if isPCDirector
        which you carefully coach,
    *if isPCDirector = false
        coached carefully by ${directorFirst} ${directorLast},
leaps straight to the ${role1Adj} heart of the character,
and
it's going to be a delight to
*if (numChars = 2)
    *if (isDoubleCast)
        see the footage intercut with the shots where
    *if isPCCastRole2
        you're playing the ${role2Adj} ${role2Job}, too.
        *goto nextSentence
    ${role1HeShe} plays the ${role2Adj} ${role2Job}, too.
    *goto nextSentence
    *if isPCCastRole1 = true
        play scenes with ${role2First} ${role2Last}'s ${
            role2Adj}
        ${role2Job}.
        *goto nextSentence
    *if isPCCastRole2 = true
        play scenes with ${role1HimHer} later on.
        *goto nextSentence
    see ${role1HimHer} play scenes with ${role2First}
    ${role2Last}'s ${role2Adj} ${role2Job}.
    *goto nextSentence
see the footage when you watch the dailies tomorrow.

```

Figure 3.9: ChoiceScript code controlling a sentence describing the first day of shooting in *Hollywood Visionary*. The complexity of the code is a sign of the strain to give a choice graph structure with templated lexia a sculptural aesthetic.

was successful: players would frequently tell me about (or mention in reviews) the movies they had made, or express appreciation that their character wasn't easily summarized by a couple of form fields.¹⁴ *Visionary* demonstrates how non-expressive input can successfully become expressive across a series of interactions that preserve the player's input and use it to build something original, and how sculptural fiction design thinking can influence games driven by more conventional narrative logics.

3.4 Challenges with Sculptural Fiction

Established storygame modes such as adventure games or choice-based fiction have built up layers of design wisdom, strategies for content authoring, and conventions understood by players and authors. As a fundamentally different mode, sculptural fiction has its own unique challenges and requires its own solutions. While we do not have the same rich body of works and theory to draw on as these existing modes do, we can consider the challenges in both creating and playing the sculptural fictions mentioned above (and their close relations discussed earlier) and begin to sketch in some rough guide posts for future creators.

In the sections below, I will identify some problems with authoring and playing sculptural fiction, and discuss both real solutions I have used on existing projects and hypothetical solutions based on future work. For authors, these problems include understanding the authoring space and keeping momentum when writing a set of

¹⁴I've written more extensively about my approach to gender and characterization in *Hollywood Visionary* in (Reed, 2015).

disconnected fragments. For players, they include orientation, tracking connections between lexia when those connections are not explicit, and addressing the loss of immersion brought on by a more detached perspective.

3.4.1 Problems for Authors

3.4.1.1 Understanding the Authoring Space

When creating linear written content, authors traditionally have two spatial aids: their position in a document and that position's context. A linear document like this dissertation is one-dimensional. The position is the particular word, sentence, section, and chapter the reader's eye (or author's cursor) might be placed at. My text editor, for instance, reveals this with a blinking vertical line and a visible scrollbar on the document's right edge. The context is maintained by showing text before and after the cursor. As I edit this section, I can see the hundred or so words before and after my current position.

In (1.2.7), I described the choice graph structure which represents, for instance, a multicursal story created in Twine (Klimas, 2009). The Twine editor shows analogous context to what you get in a linear document editor. One lexia might be active, represented by an open editing window displaying its content. Its context—the lexia that link to or away from it—can be seen in the form of the edges arriving at and departing from the selected lexia, represented as a box with connected lines.

With sculptural fiction, however, the situation is more difficult. Neither at author time nor play time is there necessarily the same concept of a “current position.”

Context is created during play by the mechanic of adding or removing lexia or connections to the story. While there are many programs for editing or viewing linear text, and a handful of options like Twine for graph-based texts, no purpose-built software exists for writing in or visualizing this kind of environment. This makes understanding position and context more difficult for both authors and players of sculptural narratives.

For example, when visualizing a graph, it's easy to either see (or programmatically verify) whether all nodes are reachable. The language ChoiceScript, for instance, comes with a suite of testing tools, one of which (called *quicktest*) will recursively check links from the beginning lexia through every possible connection. After finishing, it flags any nodes which have no paths leading to them, presenting these possible bugs in a list to the author.¹⁵ But again, sculptural fiction has no preexisting tools for this.

In addition, when authoring along a one-dimensional spectrum it's possible to “get in a groove” and write thousands of words in one sitting, moving forward through the unfolding story along with the protagonist. With a pre-planned graph structure, a writer may similarly write through one version of a scene in one creative burst, then go back through and add other branches later. As they go, they can reread the sections leading into the new branches to reestablish context for each particular moment.

But lexia in a sculptural fiction are resolutely singular. The more sculptural they are, the less easy it is for the author to definitively sequence them, or simultaneously envision all the contexts they might appear in. Writing becomes a continuous start-stop process, and (speaking from experience) it can be hard to maintain momentum.

¹⁵See <https://www.choiceofgames.com/make-your-own-games/testing-choicescript-games-automatically/>

18 *Cadence* contains only about twenty thousand words of fiction, but each sentence was its own, independent piece of story. Writing took much more effort than a prose story of similar length.

Finally, authors might need to use special syntax or affordances for varying templated text, and it can be difficult to remember all these affordances at once: not only the correct syntax, but also the options available, whether they're being applied equally across all the text that's been written, and so on.

What sculptural fiction really needs, then, is tool support. At each level of authoring, we would like to see tool features that help give authors the same kind of context they have when editing other kinds of text:

- While editing text, a view that can dynamically compute possible earlier and later nodes, and show some of these options before and after the lexia currently being edited
- A language that allows for easily specifying variations without needing to worry too much about syntax, either because there are auto-complete and -correct features, because the format is simple enough to be easily remembered and applied, or because of GUI elements that insert this syntax for the author
- A compiler or validator that can catch mistakes made, either at author time (akin to Word's spelling and grammar checkers)
- Tools for helping ensure the variations employed are covering an interesting range of possible states

In fact, we have no such tool yet, meaning most authoring to date for sculptural fiction has taken place in text files, in Excel spreadsheets, and in hand-edited structured data formats like XML or JSON. Clearly, this is not ideal: these tools offer few or none

of the features we might want, and some are miserable environments to be creative in (as anyone ever trying to edit a paragraph of text in a spreadsheet cell can attest). While work continues towards standardization,¹⁶ in the meantime sculptural fiction authors must also be tool and system designers, as well.

3.4.1.2 Finding bad states through random testing

One component of a sculptural fiction author’s toolkit is suggested by ChoiceScript’s other automated test tool, *randtest*, which plays through a story thousands of times making random choices at each lexia. At the end, a report is generated showing how often each piece of content appeared. Content that never appears even after thousands of playthroughs does not definitively indicate a bug,¹⁷ but shines a spotlight on a potential problem with the logic or connections leading to that node. When authoring *The Ice-Bound Concordance*, we used a random tester to help detect content with preconditions that could never be reached; or conversely, conditions that were too permissive (meaning cards that should have been conditional on player decisions were coming up in every playthrough).

Random testing is an established tradition in software development (Hamlet, 1994) that often proves useful in discovering bugs not detected by other methods (such as unit testing, which assumes programmers can predict in advance all the ways a piece of code might fail). Especially when a combinatorial space is too large to test completely,

¹⁶I discussed some of the context and history of this work in (2012c).

¹⁷One could imagine a puzzle that involves entering a six digit combination found elsewhere in the game, for instance: only one out of a million random playthroughs would be able to use brute force to get past this bottleneck.

random testing can discover subsets of that space with problems. (For instance, one such subset might contain every possible set of actions that involves a particular lexia with incorrect conditions.) While not as immediate as context provided directly while authoring, random testing for sculptural fiction provides a tool for discovering problem areas. An automated tester that uses a basic player model (Charles and Black, 2004) to replace random choices with something closer to the way players might actually make decisions in a given storygame would be even better, potentially focusing the search space on the possibilities most likely to be seen, and reducing the search time required to find useful insights.

Random testers need a way to distinguish good cases from bad. Most digital tests for correctness simply return a binary pass or fail when run. When authoring a creative artifact, however, context tends to be more complicated. We might need a more complex heuristic¹⁸ for identifying problem spots in a sculptural fiction. Are there ways we can provide this sort of context for a sculptural fiction?

We might propose a sort of “story integration testing,” riffing again off software development terminology for testing how a whole system behaves, rather than considering its parts individually. Story integration testing could describe undesirable narrative possibility states to the tester, and let it flag playthroughs where these situations arise, adding to the list as we discover more ways lexia combinations might be problematic, given the structure of our system. For instance, in various hypothetical systems, we

¹⁸In computing parlance, a heuristic describes an algorithm via which code can make an approximately correct decision that works in the majority of cases, without needing to fully consider or simulate all the variables for an absolutely correct answer.

might want to flag a story as bad when:

- its tension rises above 10
- it ends in fewer than 3 nodes
- it lasts longer than 100 nodes
- it contains the text “a elephant”
- it contains a noun phrase where the article does not agree with the noun
- the character of the Narrator never appears in the story
- it includes more than three characters with tag “villain”
- it never makes use of the “demonstrateCharacterPersonality” template
- a character appears in a node chronologically after that character’s death¹⁹

The given rules could be binary flags, where matching a certain number triggered a playthrough to be flagged; or they could be weighted, with a particular threshold required for them to be brought to the author’s attention. As problematic combinations appear, the author can correct them, continuing to re-run tests at frequent intervals. Some of my colleagues at UC Santa Cruz have done fascinating work into formally specifying correct game states (Osborn et al., 2015) and in using actual playtraces from a storygame to gain insights into the quality of the stories being produced with it (Samuel et al., 2014). As more nonlinear stories are written with novel structures like sculptural fiction, an infrastructure for author and support tools will hopefully begin to take shape around them.

¹⁹Of course, these rules would be specific to a given storygame: in a story with flashbacks, a character appearing after their death might be perfectly acceptable.

3.4.2 Problems for Players

3.4.2.1 Orientation

The flip side to authoring complexity is navigation complexity for the player. This is a problem even for static multilinear narratives, and the disorientation problem in hypertext that Landow identifies (discussed in 3.2.2) may be even worse for sculptural fictions, where the structure itself, instead of or in addition to the player's position within it, is changing. Players may have less guidance: there may not be a fixed node from which to start, or a limited set of option from which to choose. How do players know when they are finished, and maintain momentum as they go?

The term “overchoice” (Toffler, 1990)²⁰ has been used to describe a situation where a choice between many similar options causes stress. Our brains may not even be wired to deal effectively with choices more complex than “take it or leave it” (Bryan, 2013) which some game designers have used to justify limiting the number of choices available at any one time.

On the other hand, Landow (2009, p. 91) also quotes scholar Morse Peckham on the positive aspects of disorientation as the basis for many works of art, which in turn act as rehearsal for situations in which navigating strange experiences are vital for survival. And for most of us, creative activities like playing with Legos are intrinsically motivating and enjoyable, not stressful or disorienting (Wellhousen and Kieff, 2001, p. 122). Game designer Ernest Adams identifies “screwing around” as a natural behavior

²⁰Also called choice overload, or the paradox of choice.

in games, which enables a freeing sense of disregarding the normal strictures placed upon play (2013, p. 99).

When, then, does an experience cross from being stressfully overwhelming to playfully creative? How can a designer encourage an experience to feel more like block play than choice overload? Three design moves are particularly useful: giving players (1) clear guidance, (2) immediate feedback, and (3) reduced consequences.

We've already discussed (3) in the context of sculptural fiction's low-cost, reversible choices, but (1) is a new concept. Guidance has been studied in games in the context of implicitly communicating goals and desired behavior to players (Wardrip-Fruin et al., 2009). In an influential GDC talk, Scott Rogers (2009) traced for level designers a half-century of wisdom taken from Disneyland crowd control, such as having large landmarks that assist navigation and provide goals; using lights and shapes to guide movements; and including unadvertised or alternate paths that let people feel clever as they discover new ways to navigate the space. While sculptural fictions may not use physical movement in the same way as games in a 3-D simulated world, designers might use comparable techniques to provide guidance for assembling stories, without explicitly limiting player options. Properly guiding players towards productive actions can be seen as a form of curating agency, carefully pruning it in useful directions rather than encouraging its reckless proliferation.

For instance, the ability to browse and share stories in *18 Cadence* (a mechanic added after its initial prototypes) gave players an implied goal: making a story worth sharing. This helped situate a potentially directionless experience within a particular

context. Games like *Prom Week* (discussed in 5.3.1) and *The Sims* (2000) also give players high-level goals, while letting the process of figuring out how to achieve them happen through exploratory play.

Immediate feedback (2) combined with reduced consequences (3) also helps reinforce the pleasures of experimentation. *Ice-Bound*, for instance, immediately shows the consequences of activating a symbol by revealing the events and endings the new combination has enabled. Players may just as immediately change the state back by deactivating that symbol. Another technique is the visual cues the game uses to show when templated lexia have text selected because of a particular game state. This reveals to the player another way in which their choices have helped shape the narrative. *Ice-Bound* also provides guidance (1) via the conversations with its central character, contextualizing the sculptural gameplay and pushing the player towards completing a story and resolving it.

3.4.2.2 Fires in the Desert

Most narrative games are modeled with dramatic unity of time, remaining with the central character throughout their journey through the story world.²¹ When an adventure game character walks off the edge of one screen and appears on the opposite side of another, for instance, it's assumed the two areas are either directly connected, or at least nearby. This is often enforced with visual cues connecting the two screens, or even creatures following the player from one screen to another. Similarly for choice-based

²¹This pattern does not necessarily hold in hypertext fiction, where connections between nodes may not represent choices and there may not be an expectation of unity of time and space between nodes.

fiction, Mawhorter (2014) identifies a storygame choice as consisting of three parts: the framing that sets the choice up, the actual options presented to the player, and the outcomes associated with each of those options. There is also connective tissue between choices, leading from the outcome of one to the framing of the next. This connecting material provides narrative context for players making storygame decisions.

For a sculptural fiction to include this context, however, it would need to be prepared to narrate connections between any two lexia that might be sequentially arranged, an authoring burden growing exponentially as the square of the total number of nodes. Perhaps for this reason most existing sculptural fictions do not explicitly model these transitions. The creators of StoryNexus describe this pattern as “fires in the desert” (Arendt, 2010): the system describes interesting episodes or highlights in the career of the protagonist, and the player is expected to use their imagination to fill in the details of how their character got from one episode to another.

Some designers have embraced this empty space as a boon to interactive narratives, rather than a weakness, engaging the player’s imagination in a way that invites them to participate (Barlow, 2016). Indeed, the gaps can be even more powerful than the pieces themselves. If I tell you to imagine a doctor driving a Porsche in Hollywood, and another doctor driving a Volkswagen Beetle in Tokyo, you might start picturing very different characters and narratives around them, as your brain begins working to connect the three elements in each scenario together.²²

“Fires in the desert” may be an effective design solution, but it’s worth noting

²²This story paraphrases an anecdote told by interactive story scholar Brenda Laurel at a 2015 seminar at UC Santa Cruz, which she in turn credited to AI theorist Roger Schank.

that it’s a technical punt. While the notion is similar to concepts from other arts such as Scott McCloud’s term “blood in the gutter” to refer to action that happens “between” the panels of a comic (1994), or the elision of details that has been called dramatic compression (Murray, 2011), in these cases the author is aware of the two disconnected nodes and can use the tension between them for intentional dramatic effect. To my knowledge, no released storygames have intentionally used a frisson between disconnected nodes this way, which given the power of the technique suggests an intriguing direction for future work.

3.4.2.3 Loss of Immersion

In my master’s thesis (2011b) I discussed the tension between the transporting joy of immersion in an interactive story (Murray, 1997) and the different pleasures of coming to understand such a story’s space of possible narratives through replay and reflection (Mateas, 2004), which seemingly requires a more detached viewpoint that sacrifices the pleasures of immersion. Immersion and replay can be seen as two apparently competing potential sources of engagement with a storygame.

Sculptural narratives would seem to harm immersion, since their primary mechanic breaks the fourth wall²³ to manipulate the story itself. Understanding through replay becomes not a mode of engagement but the central mechanic. This centralizing of replay perhaps points to a solution: the deadening aspects of multiple playthroughs are lessened, since the possibility space can be explored in a single play session. Making

²³Except in a case where this shifting of reality is worked into the fiction of the storygame, such as in the metagaming mechanics discussed in (3.2.3).

that exploration the primary mechanic suggests the player might become immersed in the act of replay itself. In their book about children's block play, Wellhausen and Kieff (2001) posit that children building with blocks enter of a state of flow, the term often used in game studies to describe an ideal middle ground where challenge is great enough to overcome boredom but not so great as to be painfully frustrating (Csikszentmihalyi, 1996). This suggests players can become immersed in the act of building stories as much as in the act of reading them or of making fixed decisions.

A higher, more disconnected perspective does pose a danger of harming immersion. One possible solution is to wrap the sculptural mechanic in a frame story, which can connect to immersion and other positive elements of traditional narratives and storygames. This is the approach taken by *Ice-Bound*, where the player takes on a role (of editor) that justifies their manipulation of the interior story. Another approach is to focus less on the story constructed so far and more on the present moment, keeping the player immersed in their character's current circumstances. This is the approach seen in *Fallen London*, where the player has no access to previously explored story nodes, just their current situation. This approach lends itself to more picaresque, episodic stories less strictly about narrative cause and effect or a developing character arc.

Another element of immersion is continuing narrative progression, and this can be difficult to create when nodes are not positioned in a fixed structure. A frame story might also help this problem by making new nodes available at frequent intervals, or allowing the player's exploration to unlock new pieces. A similar approach is taken in *Her Story* (discussed in 2.5.3), which gives players a particular starting node, and then

lets them explore in any direction they choose, occasionally stumbling across nodes that suggest a new wave of potential exploration. This is a kind of structural encoding of *Pale Fire*'s "rhythm of successive discoveries" where new discoveries cause the reader to rethink and recontextualize everything they've encountered before.

3.5 Futures for Sculptural Fiction

As discussed previously in the chapter, sculptural fiction's foundations are not all new ideas. In Eric Zimmerman's analog game *Life in the Garden* (2000), the player draws cards from a deck of evocative statements and places them within a card-sized book containing opening and closing sentences, creating a temporary story that sometimes has serendipitous resonance. "The reader," Zimmerman writes, "stumbling across these narrative fragments, invents ways to connect them, imparting to them additional meanings" (2007). Zimmerman predicted the experience would feel robbed of its power were it implemented in digital form, and, like David Cage, calls it a "magic trick." In *Garden* the magic is a ritual act designed to imbue random juxtapositions with significance.

In fact, ten years later many simple story generators we see online (such as Twitter bots using expansion grammars, or my own piece *Minimalist Story Generator #1*, discussed in 8.3) rely on similar techniques. While entertaining, these works tend to provide no more than a moment's diversion: not the more sustained pleasures we expect from both stories and games. For sculptural fiction to make a claim as a legitimately

new mode of storygame, it must embrace the strengths of a digital implementation to move beyond the simple pleasure of random juxtapositions. Below I outline two avenues for future work on this front.

3.5.1 Supportive Play Partners

Wellhousen’s work studying block play in the context of a nurturing teacher-student relationship suggests a more active role for a sculptural fiction system in curating blocks that might be of interest to players. Many of the techniques described for being a “supportive play partner” in block play seem ripe for adaptation to a sculptural fiction system, driving an engine we might call a “sculptural manager” (comparative to a “drama manager” for a graph-based interactive story). For instance, here are paraphrases of some of Wellhousen’s tips (2001, p. 84) for teachers using block play:

- Arranging a simple configuration of blocks and inviting the child to copy your formation; or reversing this and copying a child’s construction yourself
- Allowing the child time to explore and play with blocks on their own, without your intervention
- Asking questions about the child’s constructions
- Observe what elements the child seems interested in, and finding ways to help them make structures that make further use of those elements
- Introduce a new block or play piece only when they’ve lost interest with their current set
- Respond to the child’s construction positively, without using judgment words like “good” or “bad”
- Engaging in parallel play, encouraging the child to ask questions about what you’re building

A sculptural manager might adapt many of the techniques suggested above for being a supportive play partner to produce an experience that helps the player maximally enjoy the sculptural mechanic through analyzing and commenting on the player’s work, providing counterexamples for them to study, suggesting pieces they might have overlooked, or intelligently limiting and granting access to pieces.

3.5.2 Flexible Nodes

The sculptural fictions discussed earlier in this document both employ a second technique easier to implement digitally than in analog form: altering text of story nodes based on dynamic context. *18 Cadence* features a simple example of this, changing a character name to a pronoun when two nodes about the same person are moved close together. *Ice-Bound* uses a richer templating system to cast characters, employ props, and alter text based on what other story nodes are active and what story themes the player has prioritized. What other ways might a more sophisticated sculptural fiction system use to change a templated lexia’s content in response to its position in the constructed story?

We might employ additional techniques that have been previously used in generative text to increase the possibility space for the rendering of any individual node. One small step towards this can be seen in my project *Almost Goodbye* (discussed in 8.5), in which “satellite sentences” establishing context and controlling pacing are procedurally inserted into a hand-authored story, based on the context created by a small number of player decisions. For instance, the system might insert “For a moment, she was quiet”

between two snippets of dialog on one playthrough, but on another, decide to remind the player where the scene was taking place by adding “The setting sun shimmered on the waves, bathing our faces in light.”

Satellite sentences are definitionally distinct from the main content of a node, and thus easier to generate than sentences that need to understand more about the structure of a story. However, it’s reasonable to propose that there might be other kinds of sentences that could be generated or varied without needing to solve all the problems of natural language generation. An engine for quoted dialog, for instance, might be passed a speaker and the words to say, and return those words wrapped in appropriate dialog markers and a speaker tag perhaps varied based on state information about that character. A straightforward example might be:

say(Katrin, “What do you mean?”)
resulting in
“What do you mean?” Katrin asked breathlessly.

...in which the fact that Katrin’s current state is “exerting herself” is reflected in the text. (More advanced systems might employ better writerly techniques than appending adverbs.)

More mechanical aspects of prose such as the subject, tense, speaker, and so on can be programmatically varied. Nick Montfort’s IF language Curveship (2009) was an early system applying these techniques to interactive fiction narration, and more recently the language Inform 7 now allows authors to alter similar descriptive aspects of its generated text. Further work from Montfort’s lab (e.g. 2013) has explored more

significant variations of text based on aspects of the speaker, narrator, and authorial intention.

Systems for using generative text grammars such as Tracery (Compton et al., 2015) might also be used as a more aggressive complement to mix-ins. Rather than a hand-encoded establishing sentence in an *Ice-Bound* lexia, for instance, we might imagine a grammar chain like this:

Hand-authored version: The long polar nights had grown aggressively grimmer.

Grammar and expansions: \$EstablishWinter ->\$LongMidnight
(might also have selected: \$ConstantCold, \$TerribleLoneliness)

The \$durationLong \$cold nights \$became \$worse. ->

The long polar nights had grown aggressively grimmer.

Alternate: The endless arctic nights got harder and harder to bear.

Indeed, *Ice-Bound* featured many places where we needed a sentence like this: replacing all of these with a system-level request for a matching sentence would have focused the authoring to the aspects specific to the particular lexia, and allowed us to potentially vary these sentences based on context (i.e. with a rule that made these sentences more or less grim based on whether particular themes like *despair* or *self-sufficient* were active).

Of course, the power to vary language based on system state comes perhaps with an accompanying loss: complete authorial control over the sound and shape of the final sentences. We might imagine teaching generative systems concepts such as assonance and rhythm, though, and perhaps even employ an approach of over-generating and then testing for the kinds of language the author wants to encourage or discourage. A computer can easily generate a thousand variations on a sentence, which another

piece of code might then choose between to find the best match, given other authorial considerations.

The “fires in the desert” problem of connective tissue might be addressable using similar techniques. We could imagine a system that generates transition text between two lexia based on active themes. For instance, adding a preface transitioning between emotional tone, such as “Despite the grimness, there were occasional moments of levity.” Here the two parts of the clause are generated in response to the dominant theme of the prior and current fragment, and the word “despite” is used because those two themes are in different classes (generally sad versus generally happy).

Sculptural fiction is most effective when leveraging the strengths of a computational medium to smooth rough edges between fragments and encourage the player towards a more personalized and rewarding sculptural experience. The success of *Ice-Bound* and other sculptural works will hopefully encourage other authors to explore making storygames in this exciting new space.

Chapter 4

Sculptural Fiction Case Study: *The Ice-Bound Concordance*

The most significant storygame I have built using sculptural fiction concepts is *The Ice-Bound Concordance* (2016; with Jacob Garbe).¹ Building on design work in earlier projects of mine such as *18 Cadence* (discussed in 3.3.2), *Ice-Bound*'s narrative system foregrounds all the core elements of sculptural fiction: low-cost and reversible decisions, an expressive exploration of possible stories, a system that highlights interesting potential additions, and a high-level view of a entire story.

4.1 Overview

Ice-Bound combines a digital game with a printed artist's book, which together tell a multi-layered story. The outer layer centers on Kristopher Holmquist, a struggling

¹The game was available as of publication time at <http://www.ice-bound.com>. This section is a revised and expanded version of material that first appeared in (Reed et al., 2014b).



Figure 4.1: Screenshot from the author's *The Ice-Bound Concordance*.

writer with a checkered past who leaves behind an unfinished manuscript when he dies. Published posthumously, the fragmented story becomes incredibly popular. Decades into the future, a wily publishing company commissions an artificial intelligence “simulacrum” of Holmquist, called KRIS, to complete the book. The inner layer of *Ice-Bound* is this unfinished novel, set at a mysterious polar base called Carina Station. The base is continually sinking into the ice, and each new owner builds new layers on top: Holmquist’s novel explored its history through a series of stories, each moving deeper into the base and thus farther back in time, each story serving as a frame for the next as people at the station become curious about the history beneath them.

In the game, the player explores permutations of these stories while carrying on a conversation with KRIS, implemented as an event-triggered dialog tree system. When the player has found an arrangement of the story that they like, they must convince KRIS this is the way the original Holmquist would have ended the story. They do this using the physical printed book, filled with early drafts, pieces of back story for Holmquist and KRIS, and strange, distorted imagery. When held up to the device’s camera, markerless tracking augmented reality recognizes which of the 80 book pages is visible. Each book page is secretly tagged with themes from a set of concepts connected to Holmquist’s fiction: the same set is used to tag the game’s narrative fragments. By finding a page that has an overlapping theme with a chosen ending, the player can convince KRIS their version of this story is “right.” As they move down to the next level, the new story will include more fragments connected to the chosen themes. The book thus acts as a high-level game controller for the narrative, slowly nudging the story in a direction more aligned with the player’s interests.

4.2 Narrative Engine

Each *Ice-Bound* story is built around a map with a set of initially unfilled **sockets**. When the story is first constructed, each socket is filled in from a library of **symbols**, lexia with associated metadata specifying what kind of sockets they might fit into (Figure 4.3). The set of symbols chosen to fill the sockets is in part random but also becomes increasingly customized to each player’s preferences as they move through

THEMES:		
• addiction or free will	• the fantastic	• going over the brink
• ambition	• fatal flaws	• journeys over destinations
• authority	• futility	• rebellion
• trial by fire	• horror	• science fiction
• cruelty	• human dignity	• self-confidence
• doing what's right	• going it alone	• self-realization
• duplicates and authenticity	• lightning not striking twice	• spiral imagery
• the fallibility of memory	• loss of innocence	• the danger of things unfinished
• the lure of exploration		
TAGS:		
• going Below	• bonding	• something monstrous
• something unsettling	• psychic powers	• a strange artifact
• mystery	• something light-hearted	• the mysterious figure
• threats of violence	• the wickedness of man	• memories

Figure 4.2: Themes and tags used in *The Ice-Bound Concordance*.

deeper and deeper stories.

Each symbol corresponds to an establishing component of a story: introducing a character trait, or setting up a dramatically loaded narrative element (such as a locked door or a loaded gun). Symbols have both a short summary and a longer text containing a paragraph or two of templated narration. Symbols are labeled from the same set of story themes as book pages, and can also be given a **tag** putting them in a more general category.

In addition to symbols, the system also has a library of **event** and **ending** lexia, with similar textual content but an additional constraint of preconditions specifying when they might appear. Preconditions are usually logical combinations of active symbols, themes, or tags. In this way an event or ending can be written to follow from a specific symbol, or to connect to an active thematic or structural concept. For instance, an event describing a violent argument might have as a precondition the specific symbol “angry” (a lexia narrating a character having this trait) or the more generic “tag threat” (matching if any active symbol has this tag, used to denote something potentially threatening).

4.2.1 Symbol Selection

A story is defined with a set of unfilled, typed sockets positioned on a map; a cast of characters for this story; and a limited stock of **lights** which the player can put in sockets to activate them. The map is used to orient the player: each story is set on a different level of Carina Station, and the position of sockets on the map indicates whether they relate to a specific character (by being placed in that character’s living quarters) or a more general narrative element (for those positioned in other rooms of the station).

Sockets are filled by symbols when the story is constructed, as explained above. Each socket has a type, which limits what kinds of symbols may be assigned to it. Sockets may request (1) a specific symbol, (2) any symbol authored for this particular story, or (3) any symbol in the whole content library. Authoring a story involves creating a set of story-specific content (symbols, events and endings), but also including global

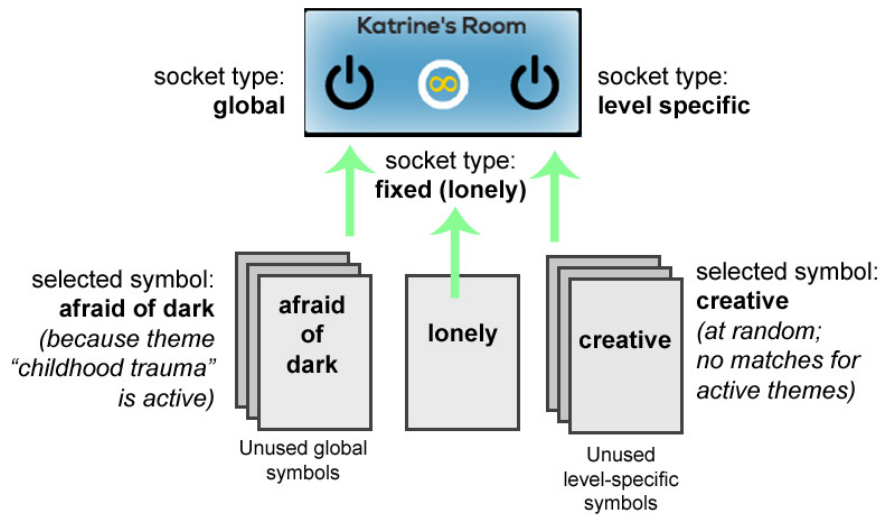


Figure 4.3: *Ice-Bound* sockets.

sockets on the map that can be filled with global symbols, written such that they could appear in any story.

Some sockets can be designated as **fixed**: permanently active, rather than switchable by the player. These sockets usually request a specific symbol and establish foundational aspects of a story or character. For example, in the second story of *Ice-Bound*, the character of Katrin always has the trait “determined”; this is a set part of her characterization the player cannot change. The combination of fixed sockets and more generalizable ones allows the author to create a fixed narrative framework but still give the player room to explore variations within that space.

When the first story of the game is built, symbols are assigned to sockets randomly, assuming all other restrictions are met. As play continues, however, selection begins to favor lexia tagged with themes the player has expressed interest in (see 4.2.5 below).

4.2.2 Player Interaction

Once the story has been constructed, the player may explore it by reconfiguring their limited stock of lights (Figure 4.4) to change the set of system-selected symbols that are active. Placing a light in a socket makes its corresponding symbol active and visible to the player. As different combinations of symbols² become active, they in turn activate events and endings based on their preconditions, and these also become visible to the player. Events narrate situations that happen during the story, while each ending posits a possible conclusion based on a subset of active symbols and events. The effect is of playing around with the initial conditions of a story, represented via moving lights around a map to highlight different areas, and immediately seeing the dramatic consequences as a result. The interface presents the whole story from a single view as a series of causes and effects, rather than moving through it from an embedded position as a character within it, as in most interactive narratives. In our developer blog for the game we referred to this arrangement as “Chekhov’s Dollhouse,” riffing on the theater maxim that if you see a loaded gun over the mantel in Act One, somebody’s going to fire it by the end of Act Three (Reed and Garbe, 2016, Part Two). *Ice-Bound*’s engine makes this notion interactive, letting the player choose what elements in a story will potentially become dramatically significant by shining virtual spotlights around a set: highlighting a symbol makes it part of the logic determining what events and endings should dramatically unfold from its inclusion.

²Mathematically, a set of active sockets for a particular build of a story is an unordered combination without repetition. If a story has k sockets and provides n lights, there are k choose n possible combinations for that story. For most of *Ice-Bound*’s stories, there are between 50 and 125 possible

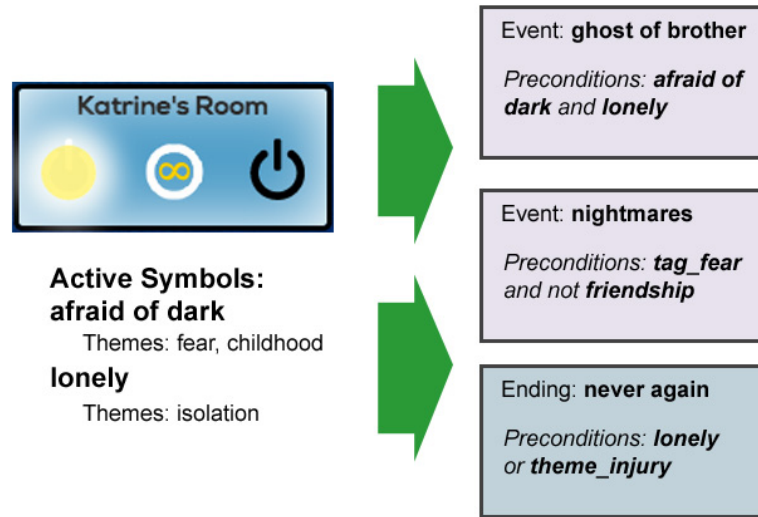


Figure 4.4: *Ice-Bound* symbols triggering events and endings.

The number of events and endings shown to a player at a given time are restricted to allow a whole story to fit on one screen and avoid overwhelming the player. Without such a culling mechanism, authors would need to worry about whether each new global event or ending might potentially add too much content to an existing level, which would reintroduce some of the combinatorial authoring burden we were hoping to avoid. Excess events and endings are trimmed to prioritize showing lexia that connect to both previously prioritized themes and themes in the current set of active symbols.

When a lexia is added to the story, the system casts it³ based on the characters appearing in the current story. A lexia specifies what kind of characters might be appropriate: specifications might be empty (meaning any character can fill this role), request a specific character, or request a character who was cast in a specific other

combinations for a given build of a level.

³In the sense of a Hollywood casting director, not the programming term *casting*.

lexia or a lexia with certain themes. A lexia can be re-cast if necessary as the player reconfigures the level. This lets us create an apparent continuity of characterization, chaining forward from symbols through events and endings, where paranoid characters are more likely to lash out irrationally, and brave characters are more likely to be cast as heroes.

4.2.3 Viewing expanded lexia text

The player can also access a second view of the story, which shows the active symbols, events and endings as a linear sequence of texts, presented as “extracts” from a hypothetical fully-written story resulting from the present configuration (Figure 4.6). The texts shown in this view have two styles of variation.

First, authors can denote sequences of random alternatives, which are displayed to the player as pieces of shimmering text that morph from one variant to another. This is to create an aesthetic effect of KRIS constantly considering many tiny variations on the story as the player reconfigures it. The player may solidify one of these texts by touching it, and keep touching to cycle through the variants.

The second way the longer lexia texts can be varied is in response to the current state: the set of active lexia, the details of the current story, and the prioritized themes. Authors can write conditional text that only appears in certain circumstances. This is discussed more in the next section.

The player can switch between these two views at will: the map view focusing on assembling and overview, and the reading view focusing on reading the results. This

```

"friendsAlongTheWay": {
  conditions: "theme_processOverProduct && tag_bonding && !
    tag_threat",
  cast: "/tags_bonding/anyone/",
  name: "_name/a/ realizes life's not about endings.",
  text: "\"What about you?\" _name/b/ said, shoving a
    crumpled sweater into _their/b/ duffel bag. _ifTheme/
    fantastical/\"After everything that's happened here,
    what's/\"What's/ next for the great _name/a/~, famous
    polar explorer?\"<br><br>_They/a/ smiled, distracted by
    _ifPersonalSymbol/a/itemNoun/the&nbsp;/(&nbsp;in
    _their/a/ hands.)/(the book about Carina Station _they/
    a/~'d brought with _them~.)/ _ifTheme/selfRealization/
    Then, realizing something profound,/After a moment's
    consideration,/ _they/a/ {put it in the suitcase, too|
    decided to leave it behind}.<br><br>\"{I'm not sure.
    Maybe|Definitely} {another adventure. Somewhere
    tropical, I think|something more normal. A nice desk
    job, maybe|some follow-up work. There are still
    questions to be answered}.\"<br><br>_name/b/ laughed.
    _timeGate/\"You said it/1950/\"You got that right
    /1980/\"Damn straight/~,\" _they/b/ said. _ifTheme/
    horror/\"Somewhere far away from this awful place, that
    's for sure. Well,/\"Well/ whatever you do, stay in
    touch, okay? _timeGate/Postcards are cheap/1997/E-mail'
    s free/2007/I've got unlimited texts, you know/~.\"<br>
    ><br>_ifTheme/goItAlone/\"Maybe in a couple years,\"/\"
    I will,\"/ _name/a/ said, _gender/a/(_gender/b/punching
    his arm fondly/rolling his eyes fondly/)(_gender/b/
    putting a hand on his shoulder fondly/putting a hand on
    her shoulder fondly/)/~. \"{You won't escape from me
    that easily|You can count on that|I will|I won't forget
    you|The world hasn't seen the last of us}.\"\",
  themes: ["processOverProduct", "selfRealization"]
}

```

Figure 4.5: Complex ending lexia from *The Ice-Bound Concordance*.

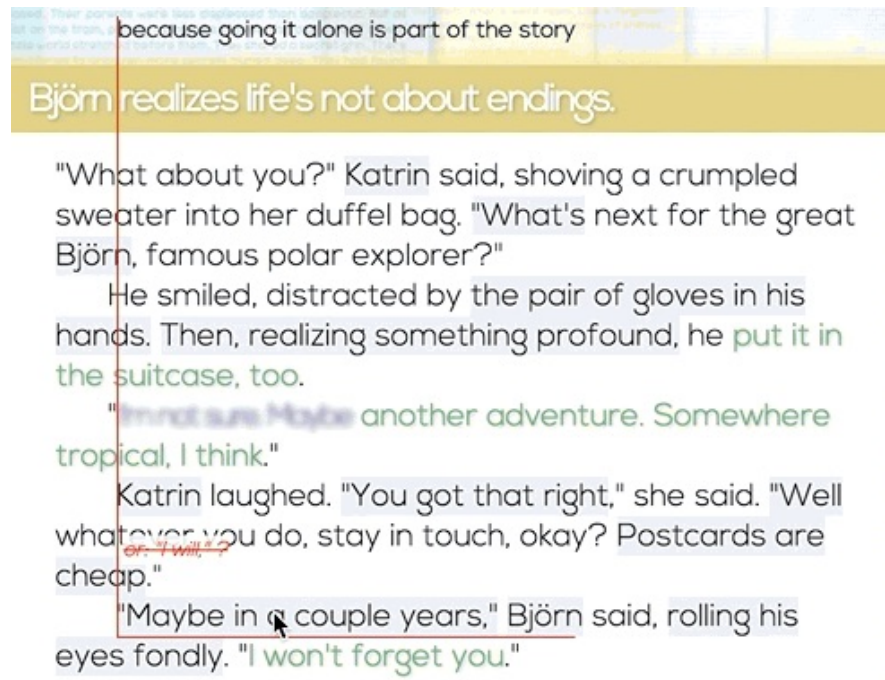


Figure 4.6: Example rendering of the lexia in Figure 4.5, demonstrating the system using a personal symbol activated by the player (in this case, the gloves) to create the sensation of dramatic closure. Text with the faint blue background was selected by the system based on the state; green or shimmering text is selectable by the player from a list of alternatives. The player can touch system-selected text to see a justification for why it was chosen.

bimodal UI helps solve the problem of including both high-level access to the overall shape of the story and the low-level textual specifics that keep a narrative interesting.

4.2.4 Template Language

The templating language was extensively used to customize lexia to the particulars of the surrounding story. A series of commands to produce text based on the story state were created (Table 4.1), from simply printing the active character’s name to more complex operations such as finding whether an active personal symbol exists for the active character and printing the name of the item associated with that symbol, or alternative text if no such match is found. Template text can itself trigger other commands recursively, or custom author-defined nodes, meaning the system also supports the full capabilities of a textual expansion grammar.⁴

My experience writing content for *Prom Week* (discussed in 6.1) convinced us of both the importance of using templating with procedurally assembled text to reestablish context (Reed, 2012a) and the necessity of appropriate tools to minimize authoring frustrations (Reed, 2011b). Templates that are evaluated only at runtime can cause hard-to-discover bugs, and an overly complicated template syntax can increase errors and pull authors out of a creative zone. For *Ice-Bound* our goal was to build a templating syntax that would be minimal and unobtrusive, removing unnecessary or hard-to-type characters such as quote marks and curly braces wherever possible. As the system grew in complexity we were not able to realize all of these ideas,⁵ but one useful

⁴While the system supports deeply nested expansions, in practice we rarely used this feature.

⁵In particular, we used just a plain text editor to create content, so could not make use of author-time

Template Command	Description
rnd	Picks a random text from list, going through all options before repeating
they	Given characters personal pronoun. Also <code>_their</code> , <code>_them</code> , <code>_theirs</code>
gender	Change text based on gender of given character
sameGender	If the two given characters are of the same gender, use first text; else second text
listChars	prints all characters on level as an English list (Tom, Doug, and Nancy)
otherChar	Prints the name of another character on the level, if possible (should be combined with <code>castSize</code> to ensure someone can be found.)
ifSymbolOn	Print one text if a particular symbol is active, another text otherwise (also <code>isEventOn</code> and <code>isEndingOn</code>)
ifSocketOn	same as above except for a socket id
ifTheme	Text for if any lexia with the given theme are active. Also <code>ifTag</code>
ifNumActiveCards	Card type should be symbols, events, or endings; operator should be <code>gt</code> , <code>gte</code> , <code>lt</code> , <code>lte</code> , <code>eq</code> , <code>neq</code>
ifWasOn	Text for whether a symbol, event, or ending is active on a prior (resolved) level
ifIsChar	Text if the given role slot is a particular character
ifHasQuality	Text if a given personal symbol is active for a particular character
personalSymbol	Requested text field for an active personal symbol for the character in the given cast position. The allowed fields are “itemNoun”, “itemName”, “traitAdjective”, “traitName” or “mapName”. Will show a generic alternative like “it” if no personal symbol is active.
lastActivatedSymbol	like <code>personalSymbol</code> , but for the most recently activated symbol. Also <code>penultActivatedSymbol</code>
timeGate	different text to show based on the current story year
castSize	Text based on whether there are one, two, or more than two cast members
mixin	Mix-in text of the given type for this character.
castMember	Prints the name of the character cast in the given position of the given card. “type of card” is symbols, events, or endings. position = 0, 1, etc...

Table 4.1: Some of the template commands used in *The Ice-Bound Concordance*.

Template	Example Rendering
<code>_name/a/ felt unsettled, misaligned. _They hunted for the threads that should have connected _them to _their life before Carina and could not find them.</code>	Katrin felt unsettled, misaligned. She hunted for the threads that should have connected her to her life before Carina and could not find them.

Table 4.2: Example *Ice-Bound* template, showing simple syntax for pronoun replacement. Fourth person (“they”) is the only one in English to use a distinct word for each pronoun.

syntax simplification was in specifying pronouns. Since most lexia were written such that various characters could play the roles within them, the system needed to adjust pronouns based on gender. Our template syntax was to use an underscore to begin a command, and we set up the processor such that simply adding an underscore before a pronoun would use the correct pronoun for the active character (Table 4.2).

This was then generalized such that any template that takes a single parameter can omit that parameter if it hasn’t changed since the last time the template was invoked. This simplified using templates in several common cases. We also added load-time verification that expansions were formatted properly as the game was loaded, allowing us to catch syntax errors more quickly.

Besides simple name and gender replacements, templates added some additional functionality, much of which was informed by my experiences on *Prom Week*. As in that game, our templates supported *mix-ins*, generic utterances that could be filled in with a more character-specific version, if available. For instance, the mix-in “swearmodifier”

syntax highlighting or other conveniences. Our inline format for commands meant the flow of them could be hard to follow, especially when they were nested; a better system might have allowed us to use indentation to visually help track this, but this was stymied by the rather pedestrian excuse of JavaScript lacking multi-line strings. (In hindsight, we ought to have written a domain-specific language for specifying this content, to avoid arbitrary restrictions like this.)

has a generic value of “in the hell,” so a character without this mixin defined might say “What in the hell are you up to?” However, we might give a more prudish character the value “in heaven’s name” instead. A stock set of frequently re-used mix-ins is a simple but effective way to make a piece of dialogue feel like it was written for a specific character, especially important in our use case where the same piece of dialogue might need to work for characters coming from different historical periods.⁶ Mix-ins also allow for more complicated references to the current story state, such as requesting an adjective that applies to a certain character (based on which personal sockets for that character are active), providing different text based on what other lexia are active, or reacting to the most recent lexia activated by the player. In all, the templating system was complex enough that we could achieve quite sophisticated dynamic story effects, such as the example seen in Figure 4.5, an ending where a character finds closure by either leaving behind or taking with them a personally significant object.

4.2.5 Resolving Endings

To move on to the next story, the player must eventually conclude their exploration of the possibility space by selecting a particular combination of lexia and choosing one ending made possible by that combination. *Ice-Bound* is not a puzzle game; there isn’t a predetermined correct ending for any story, and in fact all endings are written with the potential to be dramatically satisfying conclusions. Instead, the player’s own sense of aesthetics, informed perhaps by relevant details of Holmquist’s life

⁶Another mix-in helping with this problem let us provide different snippets of text based on what year the current story is set in, perhaps changing a laptop to a television to a radio depending on the era.



Figure 4.7: Resolving endings with the printed book in *The Ice-Bound Concordance* on iPad; the augmented reality overlay detects the page and displays altered content on the screen.

and work gleaned from the book and interactions with KRIS, guides their decision of which ending to select.

Once an ending is chosen, KRIS asks players to provide external evidence to justify it. This is done by finding a page in the companion printed book that relates to the themes of the chosen ending, and pointing the device’s camera at this page (Figure 4.7). We used augmented reality with markerless tracking (via the now-discontinued commercial Metaio SDK) to identify individual book pages from the camera feed. Each ending and each book page are tagged with themes. If the reader shows the camera a book page that has at least one theme in common with the selected ending, KRIS will agree that the ending is valid, and unlock the next story for the player to

explore. If no themes overlap, KRIS asks the player to try again with a different page. We also use the augmented reality engine to overlay images and movies on the book page, as if the player is seeing its content filtered through KRIS’s eyes: an opportunity to provide an additional layer of storytelling.

Once a story has been resolved with a particular ending, the ending’s themes are strengthened. When constructing the next story and assigning symbols to sockets, the system will prioritize symbols that have strengthened themes. As the user progresses through the stories, therefore, an increasing percentage of global sockets are filled by symbols with story elements similar to those the player has indicated a preference for, through their selection of endings and pages. As an example, if a user resolves several stories with endings incorporating elements of fear and dread, they’ll start to see more and more stories seeded with character traits or plot events related to those themes. The player is thus “completing” a version of the unfinished interior *Ice-Bound* that moves slowly towards their own preferences of what they think the story should become.

4.3 Conversations with KRIS

While the player explores the story, they carry on a running dialog with KRIS, the digital ghost of the book’s original author looking over their shoulder. This conversation happens in two modes: a full-screen conversation menu interface similar to dialog in other games, and a “conversational asides” mode on the main interface for KRIS to react to things as you play. Some asides can trigger optional full conversations

if the player clicks a presented “Discuss This” button. While conversation is sometimes triggered by preset game events, much of KRIS’s dialogue is in reaction to player choices: activating a particular symbol, selecting a certain theme, and so on.

The conversation’s dialogue system can make use of the same templating engine as the main game text, which allowed us to customized these frame story interactions much as we could the interior text. For instance, KRIS can dynamically reference themes in the symbol you just activated, or mention the characters in the current level in a comment not written specifically for that story. As with the story lexia, this customization in the conversation lexia allowed us to create the impression that KRIS is reacting to the player’s specific current situation.

KRIS also does standard conversation engine tricks, such as being able to remember which choices you’ve previously made in conversation and sometimes later respond to them, display conditional choices based on game state, and merge choices back into a single thread with seamless transition text.

4.4 Revelations

Another narrative mechanic was added later in *Ice-Bound*’s development to set up the story’s conclusion and provide more structure to KRIS’s frame story. We created a set of nine *revelations* that KRIS can have as he pieces together his back story, such as the fact that his creators have altered his mind in an attempt to make him a more efficient writer. The revelations have no dependencies on each other: they can

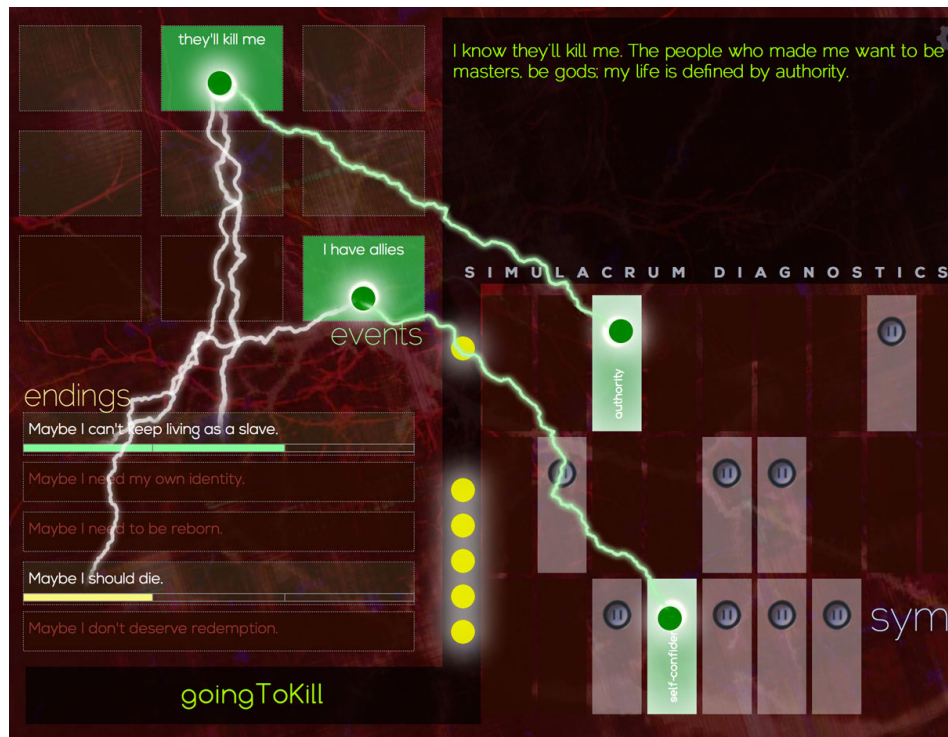


Figure 4.8: Screenshot from final level of *The Ice-Bound Concordance*.

be learned in any order. Each is connected to a number of *Ice-Bound*'s themes. If the player shows the game a Compendium page that connects (through the theme) to an unknown revelation, it can trigger a sequence where the page helps KRIS remember or understand a part of his story. If enough time passes without a revelation, one not yet known can be triggered by the player simply activating a symbol with a connected theme—something extremely likely to happen given the exploratory nature of the space and the interconnectedness of the themes.

When the player reaches the final level of the game, KRIS suffers a system failure and is rebooted into a special diagnostic mode (Figure 4.8). This mode visually represents the connections described above. All of the revelations that KRIS has learned

on this playthrough are styled as “events” that can be activated with lights. The player must “power” a revelation by connecting it to one of its related themes: doing so triggers text explaining how KRIS knows this is true because of an example he’s seen in the player’s traversal of the story: one of the lexia activated by the player that connects to this theme. Powering revelations then begins to light up one of five possible endings for KRIS, each based on a conclusion he might come to about himself and what his own story means. Each ending has three revelations as preconditions; the user must power each revelation (itself powered by a connected theme) to confirm a desired ending, effectively “building” an argument for why that ending makes the most sense. For instance, an ending where KRIS decides to escape from his captors with the help of a rebel group fighting for simulacrum rights requires activating the revelations that he has failed and been remade multiple times before; that Tethys will destroy him if he doesn’t succeed; and that the rebel group exists and wants to help him. Each of these revelations is in turn justified by scenes the player chose to make part of their *Ice-Bound* stories.

This ending sequence is radically different from the way traditional interactive stories work. Like the main game, it is entirely driven by play mechanics of exploring a narrative possibility space. The content seen is built directly from the set of lexia the player has selected for stories and themes they have chosen to emphasize, while the revelations appear in part based on which shocking pages from the printed Compendium the player has chosen to scan. If an ending is unavailable, the player can see why—they never gave KRIS the information he would have needed to reach the conclusion that

would have unlocked it.⁷ In theory, players come to a much deeper understanding of, and connection to, their chosen ending: having built up its rationalization piece by piece across their traversal and logically wired its premise together in their final act of play.

4.5 Authoring for *Ice-Bound*

Ice-Bound proved to be a more ambitious project than we initially intended. We estimate we spent perhaps two person-years in total in developing it.⁸ Given that one of the stated goals of sculptural fiction is to reduce authorial burden, a word on this might be in order. Much of the time-consuming nature of the project can be laid at the feet of its unique complexity. The game was not originally pitched to demonstrate sculptural fiction, but for an art grant involving merging digital and physical books: sculptural fiction ideas then ended up becoming a smaller part of the overall project, which involved many self-inflicted distractions such as (1) the need to design and print a full-color eighty page art book; (2) the incorporation of augmented reality, which was a developing and fiddly technology that required a lot of time to prototype and get right; (3) supporting two platforms (Windows and iPad), the latter of which puts severe constraints on designers; (4) the decision to code the game from the ground up in

⁷In order to give players the broadest flexibility, and prevent edge cases where players could theoretically have reached this level without unlocking any endings, we added a mechanic where the player can spend one of their lights on this final level to scan an additional Compendium page: reducing their currency for activating endings but allowing them to grab a “missing piece” they might need for an ending they really want.

⁸We built a prototype of the first two levels, in which perhaps half of the coding was finished, during the summer of 2013, during which time we were fully supported by a grant from UC Santa Cruz. We then worked on the project in between other responsibilities as full-time PhD students through to the final release in January 2016.

Javascript rather than using an existing game engine optimized for our release platforms; (5) doing all the UI design, art, coding, and writing ourselves; and (6) additional work from a successful Kickstarter in implementing backer rewards (which included custom in-game content and stretch goals, such as a music engine).

All that aside, *Ice-Bound* still had a fairly large scope. The final game has seven levels or stories, plus an eighth final level with different mechanics. Each level has its own set of story-specific lexia (broader than most players will see, to account for prioritizing content based on selected themes), and we also needed a large selection of global lexia to fill in the global sockets for each level. To support a story of this scope, which we estimate takes six to eight hours for a full traversal, we ended up authoring about fifty thousand words of story content. While a lot, this is still significantly less than other similarly-scoped interactive story projects I've worked on, such as *Blue Lacuna* (discussed in 2.4), which had seven times as much content for an only slightly longer experience.

One way this is achieved is that in traditional branching stories, much of the content not visited on a traversal remains forever unseen, unless the game is replayed. Sculptural fiction's affordances give players an option to see more of the content and then decide which to incorporate into their own canonical story. The content not selected is not hidden from the player, but viewed as part of the central mechanic of considering possible additions. Much as you see most of a set of Legos when you're building something with them, this implies more of the created content can be encountered by each player, meaning less effort is spent on content seen only by a fraction of players.

To effectively author *Ice-Bound*, we made a number of supplemental tools to help us understand the content we'd written so far and what remained to be created. We created two distinct visualizations (Figure 4.9) showing the possibility space for a given level (Garbe et al., 2014), one focusing on the dynamics of how the player might interact with a particular build of a level, and the other focusing on all the possible ways that level might have been built by the system. While in practice it proved difficult to keep these tools up to date as we made necessary revisions to the game's code base, they helped clarify our thinking about how level dynamics should function and pointed out problem areas where our existing authoring wasn't adequately covering the possibility space.

We also created a report viewer that would show statistics on authored content, which was especially useful to ensure the set of themes, book pages, and lexia had enough overlap with each other that they could be used mechanically (that there were multiple pages with each theme, for instance). Finally, we created an autotester that would naïvely play through the game a set number of times and gather statistics on how often individual lexia appeared and if or when the game crashed. This was useful for finding bugs in both the code but also the authoring: if a particular lexia never appeared after a hundred random playthroughs, it was highly likely its preconditions were overly constraining it such that it was impossible (or highly unlikely) for it to appear, a situation which could then be amended. The autotester also provided some validation for the assertion in the last paragraph that sculptural fiction's lexia are more widely seen by players: the autotester indicated that on the average random playthrough,

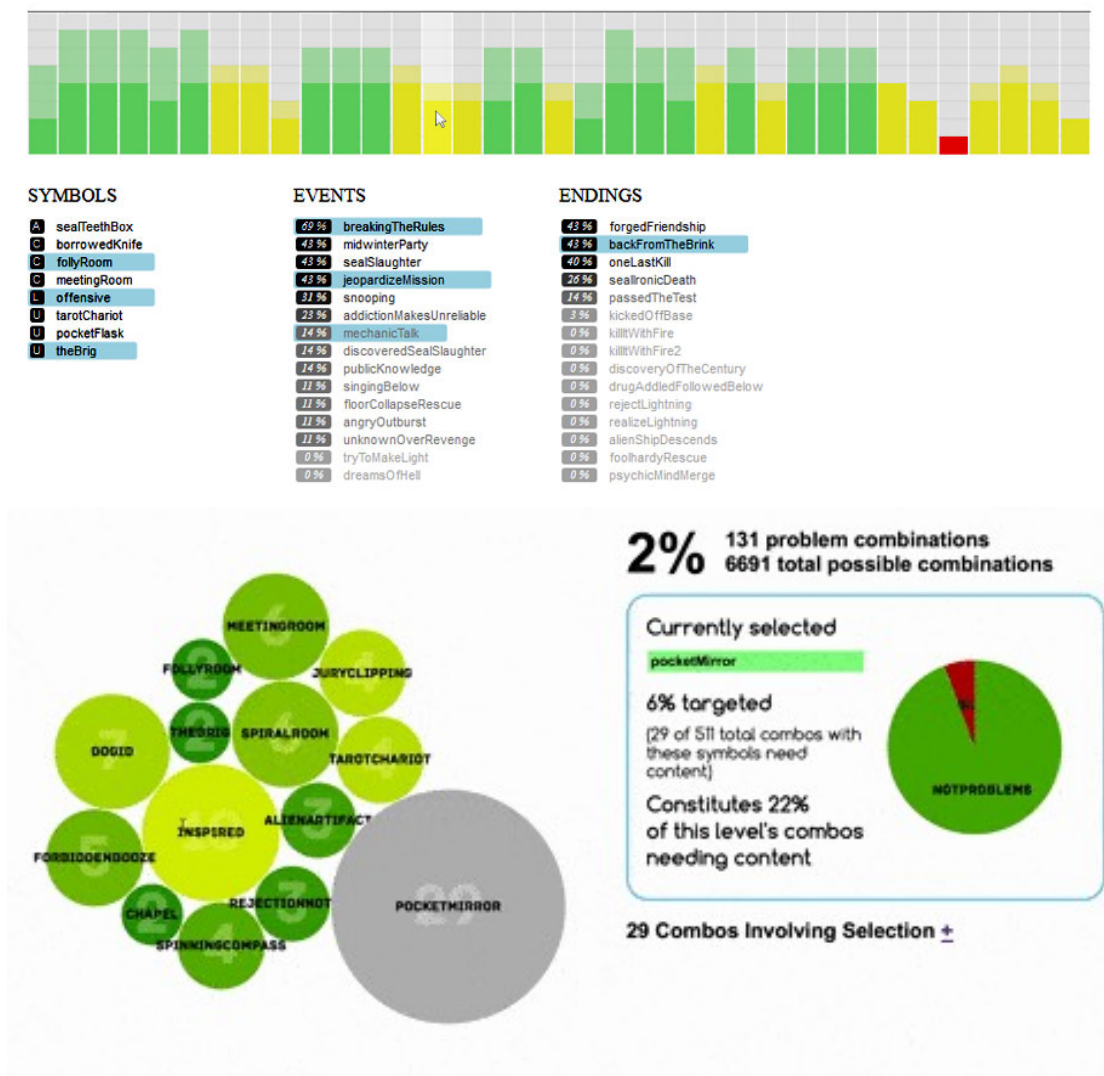


Figure 4.9: Visualization tools for *Ice-Bound*.

roughly fifty percent of all authored lexia were seen, a higher percentage than in many traditionally authored branching stories.

4.6 *Ice-Bound* as Sculptural Fiction

While *Ice-Bound* is a complicated game, and exploring sculptural fiction ideas was only one design goal, the core gameplay hews closely to the ideas outlined in the previous chapter. At a high level, the player cycles through a loop of choosing which high-level themes are important to a story, exploring a possibility space generated by the system based in part on those themes, building one potential story in that space out of the available pieces, and then choosing themes again to resolve their story. Though an individual level only has a small number of symbol lexia to directly use (between six and twelve), these can be manipulated to produce around a hundred combinations per level, and expressive input is further enhanced through (1) the continuing customization of that set of lexia to the player’s earlier decisions, (2) the much larger set of event and ending lexia created in response to various combinations of symbols, and (3) the story view which lets the player further customize the particulars of the story.

As a released game, *Ice-Bound* has been a critical success. It received the 2014 “Best Story/World Design” award from IndieCade, one of the premier festivals honoring independently produced games, among other nominations from respected game institutions, as well as reviews on a number of major gaming sites.⁹ Several reviewers have discussed the sculptural elements of the system and been excited by its potential:

⁹An index to awards and reviews can be found on www.ice-bound.com.

The deeper we travelled, the more the scope of the game impressed me. . . . The more I pressed for the story to contain elements of the fantastic, the more KRIS seeded fragments with impossible stairways, twisting libraries, figures made from icicles. . . . When I realised quite the extent to which the story was procedural and responsive, it felt like a dizzying pit had opened up. (de Quidt, 2016)

Another reviewer picked up on the constructive aspect of the game, and the potential it shows for the future of digital stories:

[A]t times, [*Ice-Bound*'s core mechanic] truly feels like writing. Removed as it is, abstracted and streamlined and simplified, you glimpse the intoxicating, terrifying possibility space of writing as you shift symbols around and watch events and endings warp in and out of existence. . . . It's the clearest example yet of a central truth we are going to have to get our heads around. In games—in any technology, and maybe even that's too narrow a scope—narrative means writing as often as it means reading. (Donlan, 2016)

Overall, *Ice-Bound*'s reception has been extremely positive, pointing a path forward for sculptural fiction as a promising technique for driving future storygames.

Chapter 5

Social Simulation

Storyteller: I'm going to tell a story.

Audience: Right!

Storyteller: It's a lie.

Audience: Right!

Storyteller: But not everything in it is false.

Audience: Right!

Sudanese ritual storyteller opening; quoted from Storytelling: Process and Practice (Livo and Rietz, 1986, p. 6)

5.1 Introduction

Many interactive story dreamers who came of age near the publication of Neal Stephenson's 1995 novel *The Diamond Age* cite it as a major inspiration on their work.¹ In the novel, a runaway girl accidentally comes into possession of an experimental book, *The Young Lady's Illustrated Primer*, designed to be a fully interactive teaching companion: weaving stories and lessons together in a curriculum designed by the book based on its perception of the reader's needs, and customizing them to the reader's

¹Ryan (2001b), Dormans et al. (2012), and Tanenbaum (2008) for a start.

culture, surroundings, and education level. While the book generates lessons and story sequences on its own, when the reader interacts with characters in the story the book secretly connects them to a human “ractor,” a kind of call-center performer tasked with bringing to life any character their terminal prompts them to play. The story is algorithmic, but the characters require a human puppeteer.

The other most-cited influence on interactive story practitioners is undoubtedly *Star Trek*’s holodeck,² in which the ship’s computer generates both stories and characters entirely on its own. These characters are presented as essentially human in their intelligence, ambitions, and emotions, and the *Trek* shows flirt with but never quite seriously engage³ the moral and ethical questions this implies. Besides the problematic aspects of creating and destroying seemingly sentient beings for entertainment, creating *Trek*’s holodeck would seem to also require solving the ultimate hard-AI problem (fully recreating human intelligence) as well as a practical question which Stephenson’s book was perhaps poking fun at: why create human-level AI in the first place when you could just hire an out-of-work actor?

Is it possible to imagine game characters that are dynamic enough to be compellingly interactive, but don’t require human-level AI or direct human intervention?

²Twenty years on, Janet Murray’s *Hamlet on the Holodeck* (1997) remains a foundational text behind much interactive story research.

³While sometimes acting like scripted robots, holodeck characters are often casually presented as having a fully human capacity for thought and also a willing subservience to be activated or deleted at any time, such as the cheerful Einstein character who Data frequently consults. The episodes that do attempt to deal with the uncomfortable humanity of holodeck characters tend to end with beard-stroking, shutting the characters down, and then never activating or speaking of them again. In the episode “Ship in a Bottle” (a sequel to the famous episode where Data challenges a Sherlock Holmes simulation to create a Moriarty that can outsmart him) when Moriarty is reactivated, he says—amazingly—he’s been aware of the passage of time while not running, and that it felt like “a lot longer than four years” since he’d last been turned on. This existentially horrifying revelation is not especially followed up on in this or any later *Trek* episodes.

Can we find a middle-ground between the largely static and scripted characters in contemporary games and the near-sentient characters in our science fiction?

Most games today use a technique so common as to be nearly invisible: to so constrain the space of possible interactions with characters that they can be represented through one or two simple systems. Shopkeepers, enemy combatants, or quest-givers are all common roles in games in part because it's believable for our interactions with them to be so limited. The action to perform with a shopkeeper is to buy or sell things from them: we can disguise a system's technical inability to handle any other actions as a social one. Jeff Orkin has studied ways to make characters perform dynamically within such highly constrained social spaces as a restaurant (2007). Poker games with highly characterized opponents⁴ and social robots playing board games (Chemers, 2013) are a few other examples of this approach. However, these characters cannot generally interact at all outside their narrow game functions, and even within that sphere their actions are typically entirely scripted, with no capacity for improvisation or potential for surprising, emergent behavior.

A more challenging approach is to represent a larger space of social interactions, and provide rules and behaviors allowing characters to act believably within it. **Social simulation** aims to create game systems within which players can experiment, strategize, and play with digital characters. As with more established game systems (such as for movement, combat, or resource management) such a social system would be fundamentally ludic. We have discussed in 3.1 how sculptural fiction addresses the prob-

⁴A small but respectable genre, explored in games such as *Poker Night at the Inventory* (Telltale, 2010) and going back at least as far as *Hoyle's Official Book of Games: Volume 1* (Sierra On-Line, 1989).

lems inherent in a static graph of lexia, giving the player more power to reposition and arrange those lexia. Social simulation instead gives both player and system more power to choose (and customize) the right lexia for a given situation, and lets arrangement happen through dynamic social rules of cause and effect.

How would social simulation improve storygames in particular? Let's imagine a game released perhaps twenty years into the future, about an epic quest—say, the one portrayed in Tolkien's *Lord of the Rings*.⁵ While many previous games have been inspired by Tolkien's books, most of them (including recent examples such as *Middle-earth: Shadow of Mordor*, 2014) focus on combat as a core mechanic: a relatively minor part of the original story. In our hypothetical game, the primary mechanic will be the social interactions within a fellowship of characters, all with different backgrounds, personalities, needs, goals, and weaknesses. While other mechanics such as resource management and the occasional fight do exist, all are radically contextualized and informed by the relationships within the fellowship.

The player might resume a session of this game by slipping on a VR headset and finding themselves at camp on a barren hillside late on an overcast morning, the characters just waking up after a long night march. Playing as the hobbit Pippin, the player can move around the camp and talk with various other members of the fellowship.⁶

Sam might share worries about the journey ahead: the player might see a chance to

⁵While theoretically Tolkien's book will enter the public domain in the year 2044, it is perhaps less believable that copyright laws will not have been changed by then to prevent this than that the advanced social simulation technology described here will have come into being.

⁶We assume text-to-speech and speech-to-text technology will both be significantly improved by this point.

console him, admonish him to be strong, or distract him by asking for a story or song. The player might bring breakfast to Boromir in an attempt to get on his good side: in exchange, he grumpily agree to some additional training in sword fighting. The player might see that the elf Legolas has gone foraging for medicinal herbs, and urge Gimli to go help him, with the hope of the two getting to know each other better and perhaps forming a friendship that might prove crucial later on. (On the other hand, leaving them alone together might cause a fight to break out—the player might take into account their current moods and how they’ve interacted so far when making this judgment.) Finally, the player might need to take care of some more practical matters: counting rations, bandaging injuries, keeping an eye open for spies or intruders. Whether these tasks are more or less important than social interactions at any given moment will be an ongoing concern.

What social simulation enables is for everything happening in this sequence to be both significant and unscripted. The player is not picking “Get Legolas and Gimli to bond” off a menu: they’re inferring from their understanding of the social simulation that sending them on a mission together is likely to have that effect. Another player might want Pippin to be the one who befriends Legolas, or enjoy a darker story where the enmity between elf and dwarf eventually tears the fellowship apart. In addition, this action and the way it’s performed is significant. Who the player chooses to talk to, the subjects discussed, and perhaps even the style used to discuss them (direct versus halting, gruff versus polite) are all noted by the system, as are the other characters’ reactions. Tone of voice, posture, and facial expressions (captured by the player’s headset) might

all be socially significant input considered by the system.

In return, the characters' performance is rich with data presenting useful information to the player. Boromir's dark glance towards Strider might signify a lingering grudge the player ought to try to heal; Gimli's slight limp a sign that his pride or distrust is preventing him from revealing the true extent of yesterday's injury; Merry's laugh, though infectious, also reveals (to a player who has learned to read the hobbit's social cues) an edge of growing panic. The group's performance in the next fight, the upcoming interaction with the leader of a desperate settlement, and their ability to solve a riddle scrawled on an ancient cairn will all be influenced by the player's understanding of and maneuvering within the group's social dynamics: to advance a story of a group of wary travelers slowly coming together and becoming a fellowship.

While some aspects of what's described above can be simulated through hand-authored dialog trees (as in games like *Dragon Age: Origins*, 2009), a system driven by social simulation provides several major advantages. First, it allows for a greater and more flexible range of emergent outcomes given the same starting state, rather than only those predetermined by the designers. Second, such a system would provide a foundation for social storytelling that allows the cast of characters to be altered or even changed entirely without completely rewriting the story content from scratch, as would be necessary with traditional techniques. The designers might allow players to explore what would have happened if Frodo had been accompanied by eight dwarves; if Legolas had been more or less haughty or Boromir more or less susceptible to the ring's power; if the wizard Radagast had joined Gandalf on the journey; if the bumbling barkeep at

the Prancing Pony had joined the party in an attempt to make up for his mistakes in Bree; if Frodo and Sam had set out on the quest alone. Each of these might produce a radically different traversal. Through this set of variations, however, a player might come to understand the themes underlying Tolkien’s work, and the game designer’s interpretations of them, in a different way. Through permutations of stories about power, courage, corruption, sacrifice, and duty, the player might gain a broader perspective on these concepts, in much the same way that playing with an interactive simulation of geography or weather can lead to different insights than a static explanation.

The game envisioned here would not replace games focused on combat, movement, or other existing game mechanics, but could open up gaming to a different audience, and allow it to more fully realize a dimension of storytelling currently under-served by game technology. While many existing storygames focus on characters, the stories they can tell are largely static: mostly the same as character-based stories in linear media. Making social interactions ludic will allow games for the first time to tell truly interactive—truly playable—stories about characters, an important maturation of the medium.

Fully realizing this vision will require three major technical advances—all of which have been made already, but will need to continue to improve:

1. Creating a playable system of social interaction for a particular story domain⁷
2. Conveying to the player the current social state and changes within it through

⁷For the *Fellowship* game, this might include notions such as friendship, honor and duty, trust, cultural background, loyalty, bravery, politeness, temptations and how susceptible one is to them, and so on.

character performance

3. Understanding in turn how the player’s performance should affect the social state

The design contribution of this chapter will be to explore how several existing games, including one co-created by the author, are tackling the first two of these problems,⁸ and generalizing these observations into a preliminary theory of social simulation storygames: what they are best at and the challenges designers face in creating them. My technical contribution is an open source implementation of a first-generation social simulation system, Ensemble, and an accompanying tool for creating content for it, both of which are discussed in the following case study chapter.

5.2 Defining Social Simulation

Simulation in gaming has a long history that touches on many genres.⁹ In the context of storygames, a useful distinction can be drawn between those elements of the game world which are *fictional* (static parts of the story world) and those which are *simulated*: dynamic, represented in some configurable way by the game’s internal logic. Aarseth (2007) speaks of the difference between a fictional game door (a fake, painted-on texture that can never open and leads nowhere) and a simulated game door, which the player can open, which can perhaps be locked, and which provides access to a new area:

⁸Though not discussed in depth in this chapter, other lines of work continue to develop the third listed problem, such as the IMMERSE project (Ferguson et al., 2015).

⁹Simulation of ballistics, of course, was one of the first uses for computers at all, driving the development of ENIAC in the 1940s (McCartney, 1999).



Figure 5.1: A merely fictional town (in Aarseth’s terms) represented by static art in *Final Fantasy II* (1988, left) and a simulated one represented by dynamic processes in *SimCity* (1989, right).

One is made solely of signs, the other of signs and a dynamic model, that will specify its behaviour and respond to our input. . . . 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. We can’t have our way with fictions, but with games, we may.

Viewed through another lens, we might speak of the “process intensity” (Crawford, 1987) of some part of a computer system: the extent to which what’s experienced by the user is the result of a dynamic, algorithmic process (as in a piece of computer-generated music using synthesized instruments) as opposed to static, pre-created data (as in a digital recording of a song). The more components of a storygame that are simulated rather than fictional, the higher process intensity such a system likely has.

Characters in storygames released to date are almost always merely fictional. Either they are not simulated at all, or they are simulated with extreme simplicity (such as by a flag representing whether they have given out their quest yet or not). Compared

to other game elements such as environment or combat, game characters have very low process intensity. In the same way the static towns of *Final Fantasy II* are fundamentally different from the dynamic towns in the contemporaneous *SimCity* (capable of growing, changing, and responding to player input), social simulation hopes to make the character interactions in a game less like static fictions and more like dynamic simulations.

Having established the difference between a fiction and simulation, what specifically do we aim to simulate? While other branches of games research consider simulating other aspects of a story world,¹⁰ social simulation concerns itself with simulating characters and, specifically, interactions between them. Developer Mitu Khandaker-Kokoris, creator of *Redshirt* (discussed in 5.3.3) defines this style of game as “anything that

¹⁰Practitioners have long disagreed over the atomic units of story, but character is frequently one of them, along with setting, plot, point of view or narration, and theme. Digitally representing or generating each of these aspects of story represents a significant branch of interactive narrative research.

Simulating the setting or world, by far the most common in released storygames, has been used as a basis for emergent stories in games including *The Elder Scrolls V: Skyrim* (2011), *Dwarf Fortress* (2006), and *Minecraft* (2011): by providing a stage and props, much as with children’s play with toys, the hope is that stories may be told on this stage, although in practice such emergent storytelling is rarely algorithmically facilitated. Giving each player a unique story encourages a practice of retelling these stories outside the game, something that rarely happens with static game stories because “any player who has played the game will have seen the exact same sequence of events in a cutscene and have little reason to read or hear about it again” (Xuanming, 2011).

Plot simulation or story generation attempts to make plots procedural by encoding some theory of storytelling, then algorithmically using that theory to combine a set of story elements into sound plots (Lebowitz, 1984; Pérez and Sharples, 2001); or by simulating character goals and planning a series of actions they might take to achieve these goals, and narrating the results as an emergent story (Meehan, 1977; Riedl and Young, 2004; Ware and Young, 2011). In some systems, this is done dynamically in response to player interaction. A sub-domain of plot generation is drama management (Laurel, 1986; Weyhrauch, 1997; Roberts and Isbell, 2008), which uses a theory of sound plotting to nudge the player towards more optimal story paths.

Point of view or simulation of the discourse level of a story is concerned with the way in which the underlying series of events making up the plot are told (Bal, 2009). This aspect of story has been less-often explored in depth than others (a good historical overview is in Wardrip-Fruin, 2009, p. 274-95) with some notable and fascinating exceptions (Montfort, 2009; Montfort et al., 2013).

Finally, simulation of theme requires both a way to understand the meaning behind narrative events and a method of generating new such events or narration driven by an attempt to convey a theme. This complex work has also been rarely explored, but has produced some fascinating results (Harrell, 2007; Zhu and Harrell, 2008).

allows social interactions with or between NPCs to meaningfully affect the outcome of a situation” (2015). However, this definition does not require simulation, and might be applied equally to a hand-authored branching path game centered around character interaction. To get closer to the dynamism behind our vision for social simulation, we need to incorporate simulation more foundationally into our definition.

We might therefore place social simulation games as a specialization of agent-based modeling systems, which have been defined as

... a collection of autonomous decision-making entities called agents. Each agent individually assesses its situation and makes decisions on the basis of a set of rules. Agents may execute various behaviors appropriate for the system they represent. (Bonabeau, 2002)

This definition comes closer to capturing the key elements of simulation useful to storygames, with high process intensity and dynamic output. We now need only add the notion of playful interaction with such a system and story-like narration, keeping in mind our existing definition of storygame (1.1.2) to arrive at a set of characteristics. Social simulation storygames:

- simulate decision-making agents, and
- allow for strategic social play with those agents, leading to
- emergent narrative outcomes.

This definition implies several changes, with cascading consequences, to the way players interact with characters in most contemporary games:

1. To enable social play at all, the game must implement primary mechanics for social interaction between characters...

2. ...and for those characters to respond, they must have rules for how their society functions...
3. ...which must be processed by the characters in such a way as to allow for surprising but sensible results...
4. ...enabling emergent narrative outcomes...
5. ...which in turn gives rise to *strategic* social play, as the player invents and tests theories for achieving a desired social state, and expressive social mechanics to allow the player to take complex and interesting social actions.

Strategic social play can therefore be seen as a dynamic arising from the mechanics of social simulation, and relying on a range of actions that allow the player to express intent within the social space.

In the following sections we will consider several early social simulation games to see what qualities they have in common, and see how some other common characteristics of social simulation storygames emerge in practice. Most notably, we will observe how the following characteristics define most social simulation storygames so far produced:

1. Authoring for most social simulations involves creating a master **social schema** for how social interaction will be simulated in this story, and two primary pools of content: **rules** connecting that schema to specific character behaviors, and **narrations** that let the system describe characters enacting those behaviors.
2. Complexity comes from **multiplicative, not additive complexity**. Social simulations do not grow outward like a choice graph adding new nodes; they become deeper, with newly added content joining a pool that enriches all interactions.
3. To encourage its reuse, **authored content is written not for specific**

characters, but for patterns of social interaction. This encourages stories with large casts, and suggests that social simulation storygames are **fundamentally about a particular social milieu** and how it enables and constraints the people within it, and only secondarily about specific characters within that milieu. This connects them to theatrical and literary traditions such as the comedy of manners, in which the focus is less on specific characters and more about how they fit into (or oppose) existing stock types and social structures.

4. Because our characterization of social simulation centers reacting to the current situation rather than long-term planning, this tends to encourage stories where **most characters passively perform the norms of the social milieu**, with the player character perhaps an exception.

5.3 Survey of Major Social Simulation Storygames

While games centering social interaction date back at least to Chris Crawford's 1983 *Gossip*, and have driven more modern games including *The Sims* (2000) and *Crusader Kings II* (2012), few released games have made both story and dynamic characters foundationally important. Some of the most significant social simulation storygames released to date are *Prom Week* (2012), *Blood & Laurels* (2014), and *Redshirt* (2015). In the next sections I will summarize each of these games, consider also several prototype systems that have not yet been part of released games, and consider the common threads that emerge, as a step towards characterizing the foundational qualities of this style of game.

5.3.1 *Prom Week*

Between late 2010 and its release in 2012,¹¹ I was lead writer and a co-designer of the game *Prom Week*, one of the most extensive social simulation games released up to that point. The details of the game’s development (McCoy et al., 2013) and the architecture of the social simulation engine behind it, *Comme il Faut* (McCoy et al., 2014) have been discussed in detail elsewhere. Here I will discuss the higher-level play dynamics of the game: readers may wish to refer to these other references for a more complete description.¹²

5.3.1.1 Game Overview

In *Prom Week*, the player controls a set of high school characters through a series of scenes leading up to their senior prom, with the goal of achieving certain social goals by the end of the week. The primary mechanic is directing a character to take an action towards another. Each action has the intent to cause a specific change to the social state. Since all actions have side effects, the player often introduces unexpected social complications, requiring changing strategies, experimentation, and serendipity to achieve their desired goals.

¹¹The game was first released on Valentine’s Day 2012; an update with a revised UI and additional story content was released on October 2012 for the IndieCade festival.

¹²Portions of this section first appeared in different form in Samuel et al. (2015a).



Figure 5.2: Screenshot of *Prom Week*.

5.3.1.2 Core Gameplay Loop

Prom Week's social rules encode tropes and behavior norms familiar from American high school movies and TV shows. These rules are considered each turn for each pair of characters to form a ranked list of **volitions**, or how each character would most like to change their social standing towards each other character. The player may direct any character to take an action stemming from their volitions (normally, only the top-ranked actions are available for the player to select). The same set of rules is consulted with the roles reversed to determine whether the recipient will **accept** or **reject** a given action. The attempted action and its result are then narrated using a templated scene (called an *instantiation*; see Figure 5.3) with attached effects that cause

“Give Advice” Instantiation #8 - Cheer Up Accept”

Intent: *Increase affinity*

Preconditions: $\{R\}$ is sad; $\{I\}$ is friends with $\{R\}$.

Line	Body	Face
I: (one of) [Hey chum... why so glum? What’s going on lately? You seem so sad. Hey... is something the matter? You’ve seemed gloomy lately.]	talking_short	concerned
R: Oh, you know. This and that. I don’t want to talk about it.	shrug	anxious
I: OK, I won’t pry. But I just want you to know that I’ll always be here for you. High-school isn’t easy, and things have been even crazier than usual with the Prom coming up.	console	concerned
R: Life has become a battleground, and my feelings are the casualties.	shakehead	sad
I: You’re a great [girl guy], %R%, a real class act! You just gotta own it! Turn that frown upside down and place a grin upon your chin. If you start thinking positive, pretty soon you’re gonna feel positive.	excited_talk	happy
R: Heh, thanks %I%. That’s good advice.		neutral
I: And most of it rhymed!	toss_hands_up (high fives)	happy

Effects: $\{R\}$ has +20 affinity for $\{I\}$; $\{I\}$ has +10 affinity for $\{R\}$

Figure 5.3: Example *Prom Week* instantiation.

some change to the social state. This change means character volitions have now altered, and the set of highest-ranked actions has now changed. The core gameplay loop is the player repeating this cycle of having a character attempt a social move and viewing the results.

An example interaction: the player directs Edward to ask Monica out on a date. Edward wanted to do this because he has a crush on Monica; because she recently did

something nice to him; and because they have several mutual friends. Monica, however, rejects this action for reasons of her own (she thinks Edward is very uncool; her friend Buzz is enemies with Edward; Edward likes samurai swords but Monica thinks they are lame). The system looks for a scene that narrates rejecting an Ask Out action, attempting to find the one most relevant to this specific circumstance. It finds a scene for the situation where the rejected asker is considered uncool. The templates are filled out for the specific characters and situation and the scene is performed, with Monica telling Edward he's not cool enough to ask her out. There are several side effects to this scene, including the asker (Edward) becoming embarrassed, and the askee (Monica) losing some affinity for him. These effects are applied, and new volitions are calculated for characters. The player now has an altered set of actions characters would like to perform: because Edward is embarrassed, he's no longer interested in trying to ask anyone out; and the few positive actions Monica might have taken towards Edward are now ranked lower because of her loss of affinity for him.

5.3.1.3 Larger Game Structure

Structurally, *Prom Week* is divided into campaigns, each of which focuses on a different character across a series of days (or levels) leading up to prom night. Each campaign begins with an identical starting state, which includes the character definitions, pre-existing relationships between them, and a pre-populated history so templated scenes can reference past events that happened before the game began (an encoded “back story”). Different characters are present on each campaign level, giving players different

opportunities to influence the cumulative social state between all eighteen characters.

Each campaign provides the player with a set of potential goals, focused on the central character for that campaign. For example, one of the earlier levels focuses on the character Zack, who has several goals for the week, including finding a Prom date and ruining a bully's chances of becoming Prom King. The player will receive an ending at the end of prom night based on which (if any) goals they've managed to achieve, but the goals do not otherwise restrict the player's action, acting mainly as a suggestion prodding the player to explore more of the system. (In particular, there is no "lose" condition: not achieving goals simply means the player gets a different ending at the end of the week.)

Goals can be satisfied through an open-ended set of solutions discovered through interaction with the characters and social state. For example, to work toward the goal of getting a date, the player could have Zack form a friendship with someone over a shared interest. This friendship itself could eventually blossom into a romance. Alternatively, since Zack's new friend has their own network of friends, his prestige will now have risen in their eyes, which might lead to certain characters who otherwise would be disinclined to acknowledge his existence now being more willing to respond. In turn, however, Zack's elevated status in their eyes might be seen as a negative by characters who dislike or have had bad experiences with Zack's new friends.

Through analogy with the popular physics game *Angry Birds* (2009), just as there is no correct way to hurl the birds to knock down the towers, there is no one way to create a goal social state through action. As *Angry Birds* is a physics puzzle game,

Prom Week could be considered a “social physics” puzzle game.

Upon completing a campaign, additional campaigns are unlocked, allowing the player to start again focusing on a different character and reset to the original social state from the beginning of the week. Further play is informed by the player’s experiences in previous levels: knowledge of the starting social state, the personality of the various characters and preexisting back story, and a growing intuition and ability to successfully manipulate the simulation.

While *Prom Week* also features a sandbox mode in which players can pick their own characters in a scene and play for as long as they like without explicit social goals, the final game emphasizes the campaign mode, which provides more guidance for players and forward momentum than the free play experience.

5.3.1.4 *Prom Week* Analysis

Prom Week was one of the first released games to demonstrate a complex social simulation working in concert with an ongoing narrative realized in full English dialogue rather than abstractions,¹³ and was recognized by the larger game community for this contribution, with coverage in the gaming press and nominations from IndieCade and the Independent Games Festival (the latter in the category “Technical Excellence”) in 2012. Through this full-scale realization of a social simulation storygame, the team was able to offer an existence proof for the genre (see 5.6 below), and begin to answer several

¹³The characters in *The Sims*, for instance, communicate using icons and a nonrepresentational babble called Simlish; Chris Crawford’s StoryTron and Erasmatron prototypes, to name another example, use either icons or a functional language (rather than a primarily literary one) with which both player and NPCs express intent and consequences.

fundamental design questions about this emerging space.

An unanswered question when the project began was whether the social simulation would be able to produce surprising, emergent stories and provide strategic play within them. By release, the team felt confidence we had achieved this goal. One traversal observed during playtesting provided a good example of this. A tester wanted to get revenge on social butterfly Monica, who had done something to upset them. Taking control over the goth couple Edward and Mave, the tester slowly introduced them into Monica’s circle of friends, who because of their social connectedness imparted on their new pals the status of “popular.” Using this new-found social leverage, the player took action resulting in the former popular clique—including Monica—losing their friendships and no longer having the social influence they originally did. The sequence culminated with Monica the cheerleader desperately asking Edward, the now-popular goth, to be her date to the prom—and Edward rejecting her for not being popular enough. The player had gained enough of an understanding of the dynamics of the social simulation to manipulate it to turn the tables, creating a delightful power-play story about one clique overthrowing another that none of the designers had anticipated.

On a more quantitative level, we gathered playtrace data across thousands of sessions with the game, which enabled us to start answering some fundamental questions about social simulation storygames. One of these questions was whether a social simulation can actually provide a unique story for each player, as opposed to being merely a complicated way of producing something akin to a branching path. For instance, if most traversals of a *Prom Week* level share similar trajectories, it could be

argued that the benefits of such a complicated system were not being realized. However, an analysis of thousands of player transcripts (McCoy, 2012) showed that within only a handful of moves, every single player had taken a unique sequence of actions and was thus highly likely to be in a unique social state.¹⁴

One unforeseen dynamic of our social simulation was the difficulty in achieving social goals that ran contrary to characters’ expected behavior (even though these were often the most interesting or narratively suggestive goals). Because the system focuses on reacting to the current state over long-term planning, and because the rules encoding a social norm are the primary means of determining actions and reactions, characters are very disinclined to perform in ways contrary to that social norm. In practice, this could lead to vicious circles where players attempting actions to reverse a social situation would simply push characters deeper and deeper into preexisting opinions and behaviors.¹⁵

To address this, near the end of development we devised a new mechanic called Social Influence Points, a resource which players gain when characters take actions with positive volition (i.e. actions they naturally want to take), and which can be spent to force them to initiate (or respond positively to) an action they would normally not consider. This lets players “build up” an ability to have characters act outside their default behaviors. Strategically saving and deploying this resource is crucial to achieving the more difficult goals. In other games (discussed below), this social pushback comes

¹⁴While every one of the billions of possible social states are not experientially unique for players, of course (i.e. it is more pedantic than useful to say *Prom Week* has billions of possible stories) this result is a promising quantitative metric suggesting that social simulation can in fact produce a wide variety of player experiences.

¹⁵As in the “Ask Out” example above, where Edward is now less inclined to ask anyone out and Monica is less inclined to accept any positive gestures from him.

from a player character who is unbounded by social norms: in some form, it seems an important tool to prevent stagnation in a social simulation.

We also made a tentative start at answering the question of how much content authoring is required to make a full-scale social simulation game. This and a detailed case study of authoring for *Prom Week* can be found in (6.1). An updated version of the social simulation engine *Comme il Faut*, called *Ensemble*, is discussed in (6.2).

5.3.2 *Blood & Laurels*

Emily Short's *Blood & Laurels* is an interactive drama set during the height of the Roman empire. The player takes the role of a poet thrust into plots to overthrow the Emperor, and based on their actions, can develop friendships or relationships with half a dozen major characters, and end up claiming the throne or helping someone else gain power. The most substantial game produced with the Versu system, a single traversal of *Laurels* can take several hours, and the dynamic nature of the system allows for many radically different subsequent traversals.

The game's surface level seems at first like a standard choice-based story, with text narration interrupted at intervals by choices where the player picks from a list of options for what their character should do or say next. However, these choices are managed by a complex social simulation, giving the player a range of ways to react to the direct actions of (and implied subtext behind) other characters. Those characters, in turn, respond to the player's actions via changing moods and opinions about conversation topics, the player, and each other. Players soon notice there are more choices than is

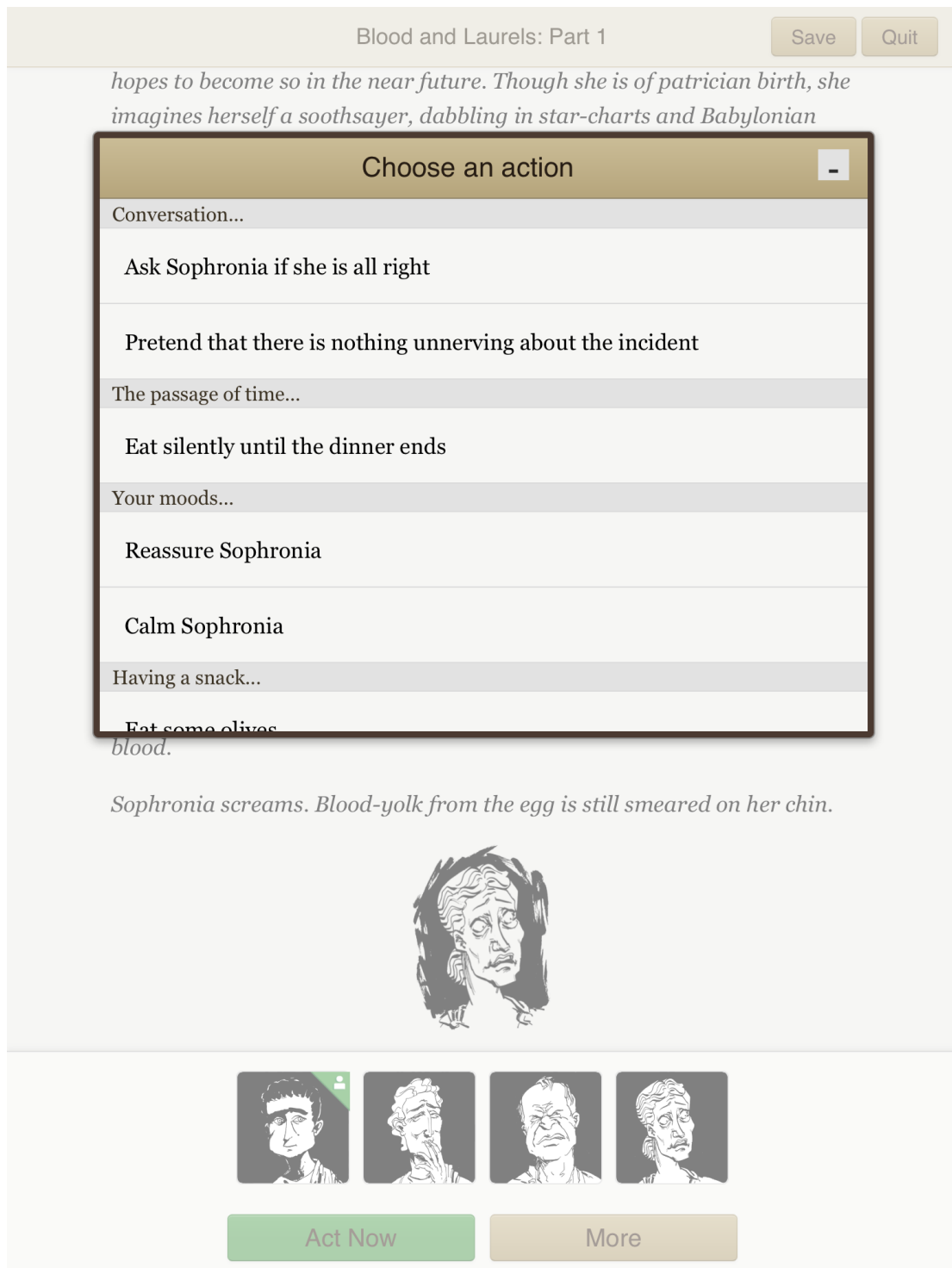


Figure 5.4: Screenshot from *Blood & Laurels*, showing a scrolling list of actions atop ongoing story text.

typical in a choice-based narrative (sometimes as many as 15 to 20) and that these choices have a higher degree of responsivity to actions.

In contrast to *Prom Week*, where each action represents a small scene of its own, most choices in *Laurels* represent smaller moments within an ongoing scene, which may contain one or two dozen individual choices. During each traversal, the player will see some subset of the roughly forty total available scenes, each of which is heavily customized to prior player decisions and character attitudes. There are nearly 5,000 possible lines of dialogue, of which only about six percent are seen on an average traversal (Nelson, 2014a).

The Versu engine, developed by Short and Richard Evans (who had previously been the AI lead for *The Sims 3*, 2009), allows authors to describe the social state through a domain-specific modal logic language called Praxis (Evans and Short, 2014). While writing in Praxis is logically unambiguous and syntactically straightforward, the difficulty of working directly with modal logic to create narrative content inspired the creation of a higher-level language called Prompter (Nelson, 2014a)¹⁶ which reportedly reduced by a factor of twenty the number of lines of code required to create *Laurels*, through inferences, auto-generation of similar code blocks and other techniques. Prompter also featured several higher-level visualization tools to help authors see how conversations clustered together within scenes, possible ways scenes could flow across traversals, and to show testing results from thousands of automated playthroughs.

Laurels drew on social schema elements built-in to Versu (it's not clear to

¹⁶Created by Graham Nelson, the designer of interactive fiction tool Inform 7, and with a similar natural language syntax for defining facts about the story world.

```

process.greet.X(agent).Y(agent)
  action ‘‘Greet’’
  preconditions
    // They must be co-located
    X.in!L and Y.in!L
  postconditions
    text ‘‘[X] says ‘Hi’ to [Y obj]’’
end

```

Figure 5.5: Example Praxis syntax.

```

(About Bluntschli and the carpet bag.)
Catherine (to Louka, naively): Captain Bluntschli! That’s a
  German name.
Louka: Swiss, madam, I think.

```

Figure 5.6: Example Prompter syntax.

what extent authors could define their own custom traits and behaviors). Those schema elements in Versu included roughly forty character traits, about 160 ways of varying a character’s speech or attitude towards others, nineteen relationship types, and twenty-six subjective numeric qualities representing feelings towards other characters. These numbers are of roughly the same complexity magnitude as *Prom Week*’s social simulation schema.

Versu was in fact being designed around the same time as *Comme il Faut* (CiF), the social engine behind *Prom Week*. While both Versu and CiF simulate characters and social conventions, and revolve around characters taking actions to move the simulation’s state towards one that corresponds with a narratively satisfying conclusion, the systems differ in several important respects.

Social practices. Characters are simulated in Versu, but also social practices: an interpersonal situation which defines appropriate actions for each role in that situation. For instance, when the social practice of *greeting* is active, a character in the “greeter” role might have the actions of “saying hello” or “effusively hugging” available: these actions may not be allowed when this social practice is not active. The Versu team describes this as a “constitutive view” of actions (Evans and Short, 2014), which differs from CiF’s approach of neither restricting nor adding actions, but ranking their desirability for a given actor and social state.¹⁷

Actions and reactions are independent. While CiF is based around actions initiated by one character and either accepted or rejected by a second, narrated as a single atomic scene, in Versu actions do not have set responses. Rather, an action causes the state to change, and other characters (including those present but not directly involved in the exchange) then have the option of reacting to that changed state with a new action.

Characters simulate the future to determine their best action. With CiF, characters gain a volition to take actions they hope will result in a better social state (as defined by their social attributes: rules for a “lonely” character might increase their volition to make friends). But CiF characters do not know in advance if their action

¹⁷CiF, of course, could implement social practices in various ways, such as by drastically changing volitions for certain actions in certain contexts. A later follow-up project by some of CiF’s creators (Treanor et al., 2016) has added social practices as a first-order feature. The need for social practices generally is also a factor of Versu modeling actions at a finer grained level than CiF, in which a social practice might be continued within the bounds a single scenes.

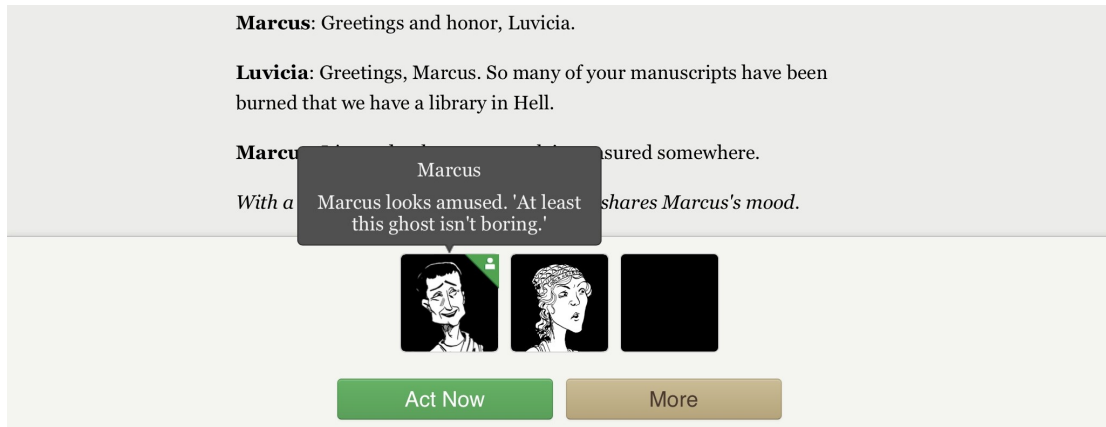


Figure 5.7: Detail of *Blood & Laurels* interface, showing character portraits and pop-up text used to convey emotional state.

will be successful. Versu characters actually evaluate how the state will change if they perform an action, giving them more accurate predictive power that can make them seem more intelligent, at the risk of making them seem too omniscient.

Characters are defined through goals, not traits. CiF defines character personalities in part through a set of traits, with rules authored to define how each trait might affect various volitions. Versu instead simulates personality through character goals defined as desirable states. For example, instead of a CiF character’s “lonely” tag and associated rules, a Versu character might simply be given the continuous goal of having at least two friends: if that state is not true, they will favor taking actions to correct that. This essentially means each character can have their own set of bespoke traits, rather than creating in advance a reusable set from which all characters must choose.

Outside of system design, *Laurels* makes a number of design decisions that contribute to the feel of the experience. While offering a large amount of potential

variation, the core of the story is strongly authored, relying on an authored structure to create an emergent dramatic experience within a coherent narrative arc. *Laurels* moves through a pre-assembled structure of scenes leading towards one of several conclusions; by contrast, *Prom Week* is less structured, providing an initial state, goals, and endings, but little support for encouraging a consistent narrative in the middle. *Laurels* exposes less information about the system’s inner working to the player, favoring immersion in the narrative over strategic manipulation of the simulation. While both games are turn-based, *Laurels*’ finer-grained simulation approach makes it feel closer to an interactive drama, whereas *Prom Week*’s lack of a specific player character or ongoing dramatic arc, and its transactional mode of social interaction, make it feel closer to a strategy game. This is also foregrounded by *Laurels*’ more active characters, each attempting to take actions that will advance their agendas, and the ability for the Versu system to support a hidden character, Fate, who can take actions to advance the scenario designer’s agenda.

Laurels and other Versu stories are also successful at producing surprising narrative results, even within a more constrained space of potential stories. I encountered one amusing example of emergence in my first Versu experience. Playing as the young woman Lucy in the Regency-era story “Family Supper,” I began flirting with the footman serving dinner. Later, when the rest of the party left the room for after-dinner drinks I remained behind, where the two of us, finally alone, could have a passionate moment that culminated in a marriage proposal. When later describing this incident to designer Emily Short, she expressed delight over this outcome and noted that she’d have to add some special cases for romances with servant characters (personal communication, Sep.

13, 2012). This quality—characters behaving in a sensible, story-appropriate way that doesn’t happen to be one the designer had particularly thought of—is one of the strengths of social simulation, and Versu provides an intriguing platform for this emergence.

However, not all reviewers had this reaction. While most agreed it felt like something revolutionary, many felt the procedurality of the system at times felt like a poor fit with the hand-crafted text (Dyer, 2014; Nelson, 2014b). This is maybe a weakness of the zoomed-in approach: it’s much more difficult for a system to stitch together a scene out of sentences than to tell a complete mini-narrative where the author has full control over the back-and-forth, as in *Prom Week*’s self-contained instantiations. Short has noted a problem with people not perceiving the procedural situations the engine was orchestrating: they only noticed the procedurality when there was a failure, and assumed all the emergent drama they encountered was actually pre-authored.

Versu, funded by game studio Linden Labs as an iPad-specific project, was unceremoniously canceled before the format could properly launch: Short had to fight to get *Blood & Laurels* released at all.¹⁸ Without engineering support, the game fell victim to Apple’s brutal upgrade cycle, and stopped functioning with the iOS 9 update in 2015,¹⁹ a huge loss to playable media’s history. Versu and *Laurels* remain a high-water mark in the ongoing efforts to marry sophisticated character simulation and high-quality literary output.

¹⁸<https://versu.com/2014/06/06/news-about-versu-and-blood-laurels/>

¹⁹<https://twitter.com/emshort/status/653206790168014848>

5.3.3 *Redshirt*

In *Redshirt*, developed by Mitu Khandaker-Kokoris, the player takes control of a new recruit on a space station, and must advance their career by making connections and developing friendships with other officers. All actions take place through “Spacebook,” a clever design decision which restricts the player’s agency to a set of familiar affordances, each with social consequences (liking posts, writing on walls, accepting or declining event invitations, and so on). The central mechanic is managing a limited number of daily actions to most effectively maintain your existing relationships, rise in the ranks, and befriend higher and higher status crew members, with the latter two goals especially interrelated.

Characters are simulated with five public variables and five private axes of “moods or current personality-related statuses.” Characters have platonic and romantic interest levels for each other, with the latter randomly increasing as the former does. As opposed to Versu and CiF’s avoidance of randomness, Khandaker-Kokoris explicitly states that “love is not deterministic”: it’s always in part a matter of chance if your attempts to romantically woo someone will succeed.

Redshirt is notable for its tight intermingling of social simulation and resource management mechanics. On a non-social level, the player must manage money, happiness, health, skills and a career path. Money is required for purchasing food (which maintains health), acquiring items (which increase happiness), and participating in activities (which boost skills). Working at your job pays a wage and also increases skills, mastery of which

SPACEBOOK

0

SEARCH



JILA GORO

TRANSPORTER ACCIDENT CLEANUP TECHNICIAN

SPECIES: ASRION

AGE: 31

GENDER: FEMALE

RELATIONSHIP STATUS: SINGLE AND LOOKING

INTERESTED IN: ALL

SEND RELATIONSHIP REQUEST

REMOVE

LATEST TOP INTERESTS: (0 MATCHING)

UNSOLICITED TINKERING

SELF-INDULGENCE

GAMBLING

MUTUAL FRIENDS PREVIEW





VIEW ALL FRIENDS

SEND MESSAGE

ABOUT ME:

"Friends can come and go, but that's ESPECIALLY true if you're friends with me."

WRITE ON THEIR TIMELINE

JILA GORO

Hate it when Asrions complain they get looked over for

FRIENDS

CHECK IN

EVENTS

ASPIRATION 1 of 3: WHAT IS THIS THING YOU CALL LOVE?

GET ROMANTICALLY INVOLVED WITH KHANI CH'ROSA

SPACEBOOK

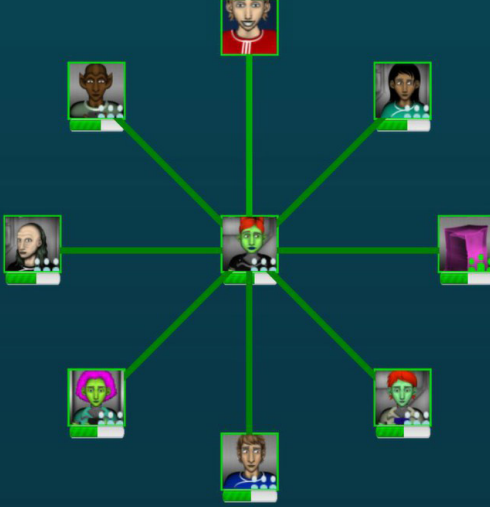
CAREER

SKILLS & INTERESTS

INVENTORY

SOCIAL MAP

SHOWING TOP 8 MOST SIGNIFICANT RELATIONSHIPS FOR JILA GORO.



MAGENTA = ROMANTIC RELATIONSHIP.

GREEN = POSITIVE RELATIONSHIP.

RED = NEGATIVE RELATIONSHIP.

VIEW TRASHA

CLOSE

Figure 5.8: Screenshot from *Redshirt*.

betters your ability to get promotions and move up a career tree into better positions.

The social simulation is tightly integrated with these mechanics, and playing with it is crucial to success. For instance, when applying for a new job, your relationship with the hiring manager, not just your skills, is a factor in whether you'll get it. Befriending people of a higher rank helps your career aspirations by raising your importance in the eyes of other crew members. Entering a romantic relationship can significantly raise your happiness, which can lead to improved job performance: but maintaining that relationship leaves you less time for extending your social network (or blatantly flirting with your boss to get ahead). In short, the two sets of mechanics are tightly coupled, with nearly every action having consequences for both sides of the game, and allowing smart players to play the two systems off each other.

The game also has some interesting mechanics to provide churn, preventing the social space from getting too static. Every so often characters are sent on "away missions" in which some of them die, upsetting the social space as friends or significant others are removed from the simulation. The ticking clock subplot also encourages the player to keep moving up the social ladder rather than remaining stagnant.

Redshirt differs from *Prom Week* and *Blood & Laurels* in several interesting aspects.

More abstracted action. Instead of attempting to narrate dramatic scenes in natural language, *Redshirt* takes the clever approach of representing the world solely as a social network: the action of expressing affinity for a character both is, and is represented

by, clicking a thumbs-up icon.²⁰ This allows for a more abstract representation and exploration of a social state, which is both easier to author for than language-based representations while also leveraging familiarity with an existing mode of social interaction, rather than needing to invent an artificial abstraction (see also 5.3.4.3 and Chris Crawford’s championing of this technique).

More strategic play. Rather than obscuring the mechanical social effects behind narrative actions, it’s usually quite straightforward in *Redshirt* to predict the outcome of a move: spending time with people or liking a post will reliably increase your affinity, assuming they’re near your rank and don’t hate you; someone flirting with you is likely to accept a relationship request; and so on. Rather than immersive performance within a sometimes obscure

social situation (as in *Laurels*) or experimentally navigating unexpected side effects and surprising results (as in *Prom*

Week), the focus here is on strategic play within a more predictable but deeply intercon-

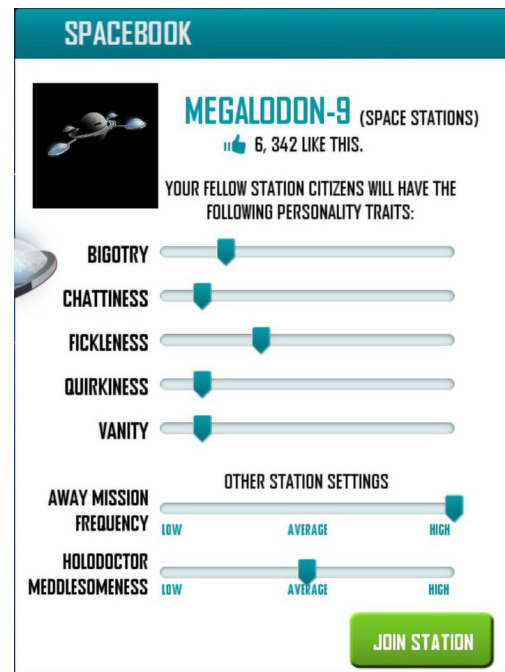


Figure 5.9: *Redshirt* uniquely allows for the generation of a new social world with specific overall qualities.

²⁰The genius of this as procedural rhetoric, of course, is in presenting a story world where there is no distinction between making real human connections and performing them on social media: the act and the online representation thereof become one and the same.

nected social space.

More emergent storytelling. While *Redshirt* reveals bits of plot as a timer slowly ticks down towards a climax, it's the purest social simulation of the three described, lacking both the strong narrative momentum of *Laurels* and the pre-planned level structure and goals of *Prom Week*. The player largely makes their own drama, creating a story of how they navigate the social state and advance through the ranks. Compounding this emergence, the cast of characters in *Redshirt* is randomly generated for each traversal. The player can even set social simulation parameters when starting a new game (Figure 5.9), adjusting how likely people in general are to be bigoted, vain, quirky, chatty, or fickle. This makes the game's stories more emergent, but less narratively stage-managed, than the other games we've discussed: its characters are less complex and interesting than the more hand-curated examples in *Laurels* and *Prom Week*.

Together, these three games provide interesting “core samples” of the space of social simulation storygames. All three simulate decision-making agents, allow for strategic social play with those agents, and provide emergent narrative outcomes, but achieve these goals in different ways. *Laurels* situates its agents within highly constraining social practices and a pre-authored story to produce emergent performances embedded in a strong narrative, while *Redshirt* gives agents more free rein to perform within a simpler system, allowing stories to arise from their interaction. *Prom Week* treats action and reaction as an atomic social unit, while *Redshirt* and *Laurels* both decouple these; and

exposes nearly all relevant information about the social state, while *Redshirt* exposes some and *Laurels* very little. It's clear that there are a wide variety of approaches, techniques, and design trade-offs to consider when authoring a social simulation storygame, and much unexplored territory yet to consider.

5.3.4 Other Works

5.3.4.1 *Façade*

I have previously written (2011b) about 2005's interactive drama *Façade*, as have many other researchers: it remains one of the most respected interactive dramas produced. *Façade* uses social simulation as a piece of a much more complicated storytelling system (Mateas and Stern, 2003), which includes drama management, natural language parsing, beat-to-beat dramatic assembly and coherence, and other techniques designed to produce an immersive, holodeck-style playable story. This kind of experience has been studied under the label interactive drama or ID (Laurel, 2013) and is beyond the scope of this section to describe: but in brief, ID attempts to solve all the problems necessary to realize our chapter-opening Fellowship experience at once (i.e. natural language understanding, real-time interruption and resumption of interrupted topics, body language, etc.) rather than tackling social interaction in isolation.

Technically, *Façade*'s implementation of social simulation is a perhaps surprisingly small part of the overall system. In the early game the relationship between the player and the two NPCs is tracked mainly by a single variable representing whose side the NPCs perceive the player to be most aligned to (the “affinity game”). More focus

is placed on interpreting the player’s actions within the context of a particular story situation, rather than giving players and characters the ability to radically change the situation and evolve new stories (Mateas and Stern, 2007).

This is, of course, intentional: *Faade* is a story about two particular characters, not just the social milieu they operate within, with dialogue and actions written for them in particular. The player is exploring the story of a particular marriage on the verge of breaking up, not the more general social notions of marriage or break-ups in the abstract. Specifically, the drama comes from existing within that particular moment in a social state, not making changes to it (and if such changes do occur, the story is over). We can thus say that from both a technical and design perspective, *Faade*’s goals as an interactive drama (immersion, dramatic specificity) differ somewhat from the goals of a social simulation (emergence, exploration of a social milieu) though clearly, the two concepts share aspirations and are highly interrelated.

5.3.4.2 StoryBricks

While never used in a released game, the Storybricks system (Bura, 2012) proposed a technique for character simulation intended to be easier to author in contexts where scalability was a concern, like large-scale online roleplaying games. The designers identified as a stumbling block the need for authoring rules to connect actions with character-defining qualities. For instance, to teach CiF how to perform a Lothario, one might create rules for a trait *seducer* increasing the volition for such a person to take bold romantic actions, and decreasing volition to reject such actions from others. As

the number of actions, traits, and other simulated social parameters increases, so do the number of rules which must be authored to describe how any given trait, such as *seducer*, influences the social schema.

Storybricks proposed to simulate both character qualities and characters' desire to take actions using the same language: a vector of personality traits, such as the Big Five personality model. A character's desire to take an action—*as well as* the action itself—are both encoded as a positive or negative vector in the set of simulated traits, such as *+3 Extroversion*, *+2 Openness*. Given this encoding, rather than creating a *seducer* trait, an author who wished to create a Lothario instead makes a personality vector pointing towards an aggressively romantic personality, and applies it to the character's behavior model. As in the Versu system, this means each character can have their own individual behavior, rather than pulling from a common set of descriptors.²¹ An interesting twist with Storybricks is that objects and locations in the world could be given these same kinds of personality vectors, affecting characters in close proximity—such as a magic ring that makes an NPC much more of a suave romantic.

Storybricks was to be used in the MMO *Everquest Next*, but unfortunately the project was canceled and the team disbanded before the technology could appear in a released game. While the conceptually simpler conflation of traits and rules is intriguing, one suspects in practice authors might have had more difficulty wrapping their heads

²¹Both systems, of course, are making a trade-off between more customizable individual characters and more reusable packages of behavior. The former can be more flexible, but requires defining an ontology which is expressive enough to support any possible set of characters the system might represent. Because the ontology is used combinatorially, this may well result in more emergent bugs that break character believability. A fixed set of traits can be more manageable, but can in contrast reduce the possibility of positive emergence (desirable kinds of unexpected behaviors).

around negative personality vectors than in authoring rules describing more directly how traits influenced actions. In addition, the simplicity means the Storybricks model by itself could not handle more complex examples of social state influencing behavior, such as rules that user to historical events to influence current behavior.

5.3.4.3 *Siboot* and other Chris Crawford projects

*Siboot*²² is the latest iteration of a thread of interactive story research that Chris Crawford has been pursuing for decades (in the most recent ones, see 1999; 2007; 2012). While not yet released at the time of this writing, the project’s public materials indicate it carries on several threads of research seen in earlier experiments from Crawford:

1. Parity between the way the player communicates with game characters and vice versa, through the use of a symbolic language for expressing beliefs, emotions, and commands
2. Characters have traits, relationships, moods, and memories of past interactions
3. Using the facial expressions of characters to represent emotional state, where reading those emotions correctly is part of the gameplay²³

The new *Siboot* also reportedly wraps the social simulation in a narrative layer where the player makes list-based choices, and connects it to a strategic minigame simulating dream combat. Crawford describes the new game as “stripp[ing] away

²²The original *Trust & Betrayal: The Legacy of Siboot* (1987) shares a back story and some design ideas with the modern incarnation in development. Another of Crawford’s experiments in this tradition, the (now sadly unavailable) game *Balance of Power: 21st Century*, is discussed in more depth in (Reed, 2011b).

²³Using character faces to provide information about internal state is a fascinating technique that has been used to varying degrees of success in a few released games, including *Alabaster*, *L.A. Noire*, and *Faade*.

half the complexity” from his earlier work. For instance, the game has only three personality traits: Goodness, Honesty, and Power (though these can have both positive and negative numerical values), and moods are expressed as temporary modifiers to these (so happiness makes someone temporarily more good, etc.) The new project also seems to be applying some useful lessons that other projects described elsewhere in this chapter have independently learned (simplicity over complexity; integration of social and traditional game mechanics; the importance of clearly communicating the social state). The project will be interesting to follow as it continues through development and release.

5.3.4.4 *Black Closet* and Hanako Games

A series of games by Georgina Bensley and Hanako Games²⁴ have explored merging game mechanics with strongly plotted narrative. In *Black Closet* (2012), the player takes the role of a student council president at an elite girl’s boarding school, and must investigate scandals to preserve the school’s reputation. The player does not act directly, instead ordering a team of minions to perform actions such as Question, Stalk, and Harass. Actions can succeed, fail, or be “too close to call” (a less severe failure) based on a random die roll and the relevant stats of the two opposing characters (which are Social, Observation, Intimidation, and Stealth). A short sentence describes the results. Several other mechanics add strategy and variety: objects can be found that increase the chance of winning conflicts, and minions can become less loyal or too stressed based on the way you use them. One minion is a traitor, intentionally losing

²⁴Also including *Magical Diary* (2011) and *Long Live the Queen* (2012).

their conflicts: this is randomly generated for each game (along with the details of the missions), and figuring out the traitor's identity is important to a successful traversal.

Black Closet alternates the strategic mission solving (visually styled and mechanically represented as a card game) with pre-authored cut scenes that serve to advance the plot and respond to game events. For instance, sending the high-society girl Thaïs on a supply closet run can trigger a scene where she complains that this is beneath her: keep doing so, and her stress will increase (eventually rendering her unusable for a day). The player is given dialogue tree choices within the scenes, which can be used to trigger romance subplots but rarely have other mechanical effects on the game.

While *Black Closet*'s characters are not decision-making agents, this and Bensley's other games in this mode are an interesting middle ground between the more computationally intensive but less scripted social simulation games discussed earlier, and the tradition of visual novels, a game genre primarily about social interaction but where little or nothing is simulated in favor of focusing almost entirely on a hand-crafted narrative. Bensley's work challenges this genre assumption by integrating strategic mechanics into her strongly plotted stories. This is achieved in *Black Closet* partly by making the mechanics of the game more about investigation than direct social interaction: other than the Minions' loyalty to (and possible romantic feelings towards) the player, no other social information is part of the system. As with *Façade*, the desire to tell a story about specific characters pushes back against a complex, emergent social simulation. The mechanics are mostly limited to influencing the individual cases that come and go, leaving the overall narrative arc mostly in the hands of the writer. These games

demonstrate another strategy for collaboration with narrative systems.

5.3.4.5 Non-narrative social simulations

While mostly outside the scope of this chapter, it is useful to briefly consider how social simulation has been used in games in ways other than to directly tell stories,²⁵ or have appeared as a component of games where most of the gameplay is concerned with other systems.

Crusader Kings II (2012) layers a social simulation and a grand strategy game of feuding European powers. Notable is that you play as a specific ruler, with a wide variety of possible traits (nearly two hundred with all the game's expansions) and extensive rules controlling how other rulers, vassals, and important people in your kingdom feel about you and each other (Kaiser, 2013). An aggregate like/dislike score for your ruler can be seen for each simulated character, along with a list of important factors²⁶ that went into that score. The player may take actions regarding the management of their country and also towards specific characters, such as imprisoning them, granting titles, and proposing alliances and marriages; both social and political actions affect how you are viewed by others, which in turn affects their actions. Some reviewers have attributed the game's critical and commercial success in part to this re-centering of a traditionally depersonalized genre on specific characters and their interactions (Smith, 2012).

Some sports management games have also featured social simulation in an

²⁵Even more outside our scope are social simulations used outside the context of games entirely, but Sun (2006) and Terna et al. (1998) are a few entry points to this domain.

²⁶See <http://crusaderkings-two.wikia.com/wiki/Traits>

attempt to model the challenge of managing competing personalities and keeping teams happy (Garcin, 2013). *Football Manager 2015*, to name one example, simulates morale for each player, relationships between team members, and a group stat called Team Cohesion.²⁷ Teammates also have seven hidden personality attributes which affect things like how likely they are to form strong relationships with each other, as well as an overall happiness score. Interestingly, while these personality stats are important for gameplay, they are not directly revealed to the player, who instead must infer them in ways such as observing body language of teammates on the pitch. The social stats are frequently influenced by gameplay decisions and vice versa, including performance of individual team members and the team as a whole, how the player’s manager character treats them in interactions, whether expectations of winning versus losing are met, and dozens of other factors.

5.4 Characterizing Social Simulation Storygames

Even with only a handful of examples, we can begin to characterize social simulation storygames as an emerging genre, and see nascent patterns and design guidelines beginning to appearing that might help in the development of future games in this space.

Specifically, we can identify four characteristics of social simulation storygames, which will be discussed in the following sections: (1) the division of content into two major

²⁷Information from guidetofootballmanager.com (2015); taken from the “Player Morale & Relationships” and “Player Personalities” pages.

pools structured by an organizing schema; (2) multiplicative, not additive complexity; (3) a structure that encourages stories about social milieu rather than individualized characters; and (4) the need for a mechanic to break characters out of social ruts, in order to ensure story-like progression.

First, however, we can note that some features of social simulation storygames are inherited from the nature of storygames. To revisit our definition, recall that storygames are:

- playable systems, with
- units of narrative, where
- the understanding of both, and the relationship between them, is required for a satisfying traversal.

We can observe that all the storygames we have discussed in this chapter feature some amount of narrative scaffolding, rather than situating the social simulation as a pure sandbox. Without *Prom Week*'s goals and progression towards a climactic prom night, without *Redshirt*'s away mission churn and forced progression up the social ladder, without *Laurels*' plot manager moving briskly through preplanned scenes, we lose out on the “units of narrative” which allow for the fundamental satisfaction of storygames, the co-mingling of those units with the underlying playable system. Social simulations without this narrative scaffolding (such as *The Sims* or *Crusader Kings II*) are still entertaining, but offer different pleasures than the storygame does.

5.4.1 Schema, Rules, and Narrations: Three Types of Content

While specifics of implementation differ, we can say in general (and specifically for the three games studied above) that social simulation storygames require authoring three types of content:

1. a **social schema** quantifying the social space to be simulated, encoding what concepts are important in a particular social milieu
2. a **set of rules** encoding how aspects of the schema connect to appropriate behaviors within that milieu
3. a **set of narrations** to communicate characters performing those behaviors

Each of these is a necessary component of a social simulation. The **schema** specifies what out of the entire possibility space of human interactions is to be simulated. Much as a particular physics simulator might consider *gravity* and *mass*, discarding other elements of motion such as viscosity or friction, a particular social simulator like *Prom Week* might include *relationships* (a binary connection type between people, like friendship, enemies and dating) and *networks* (a subjective numeric feeling directed from one character to another, like coolness or affinity) as important elements; another, like *Blood & Laurels*, might include the notion of *social practices* (particular contexts driving appropriate behaviors, like playing whist or having dinner). We mentioned earlier our hypothetical Fellowship game might include concepts like *honor*, *duty*, and *bravery*, which are less relevant to social contexts of either of the other games. The variables of interest for a social simulator vary more broadly than in a physics simulator, encoding

the aspects of human interaction each game designer feels provides the best leverage to encode the social milieu of their story as a playable system.

Part of the schema is the actions characters are allowed to take within the simulation: the ways they are allowed to alter their relationship to that schema, such as starting or ending a friendship, or deciding someone has less honor than they'd previously thought. The set of actions and the concepts they can act upon together form the social schema.

The **rules** then use the components of the schema as building blocks, defining how elements of the schema influence a character to take those actions. *Prom Week* might define that a person is less likely to be mean to a friend; *Laurels* might say that a person with the quality *romantic* would like to be in a flirtatious relationship with someone having the trait *poet*.²⁸

Rules might use one of two broad strategies to determine how in aggregate they translate into actions. *Weighted* rules each have a numeric value, and the desirability of actions corresponds to how highly they've been weighted after all rules have been consulted. *Aggregate* rulesets work the same, except each rule has an effective weight of 1: the preponderance of matching rules is what decides the outcome, like a pros/cons list.²⁹ Likewise, we might have several different strategies for selecting actions once we've consulted our rules: picking the highest-ranked one, using a weighted probability

²⁸There might be other sets of rules involved in a social simulation as well, such as *Prom Week*'s trigger rules, which define that one state (like a person being cheated on) should immediately lead to another (the character becoming heartbroken and angry at the newcomer). But a core set of rules connecting schema to actions are foundational to any social simulation.

²⁹In (5.5.6) below, some of the challenges of weighting rules are discussed as an authoring problem; aggregate rules forgo this difficulty at the expense of a lesser degree of authorial control.

curve that makes higher-weighted actions more likely to be performed, restricting actions below a certain weight threshold, and so on.

The third component is **narrations**: lexia with which the system conveys to the player that a character is performing an action. These might vary wildly from game to game, taking the form of simple status messages in *Redshirt*; templated dialogue scenes in *Prom Week*; perhaps a complex plan realized through generated text, performance, and even music in a more elaborate game. But at the least, the system must include at least one way to narrate every possible social move a character can make: one narration per action (or perhaps multiple narrations, if an action and potential consequences are treated as an atomic unit, as in *Prom Week*).

Even the simplest narration system is likely to need to make some use of templating, where the system can take a generic plan and contextualize it for the particular situation that's occurred: { *Character A* } *becomes friends with* { *Character B* } might be rendered at run-time as *Monica becomes friends with Edward*. Games might also need additional narrations to communicate the current social state (such as *Laurels'* facial portraits showing character moods, or *Prom Week's* random text messages communicating the current social state.)

This three-part system is markedly different than the way content for more familiar kinds of storygames is structured, and even from sculptural fictions or the collaborative fictions discussed in the next main chapter. The difficulties of this unique style of authoring are discussed in detail starting in (5.5).

5.4.2 Multiplicative, Not Additive Complexity

Another characteristic of social simulation storygames is that the simulation grows in complexity through multiplicative layering, not additive extension. Constructing a typical node-based storygame is an additive practice: increasing the width and height of a graph of possible traversals. While the structure grows in size, the mechanism used to navigate it remains the same. Social simulation games, conversely, grow deeper rather than broader. Each new schema element, rule, or narration joins the underlying simulation which is active at all times, further enriching the experience of each social interaction within it. When you add a notion like “honor” to a social simulation, it’s not a matter of moving to the part of the graph where honor exists: honor is now everywhere and always a part of how your characters reason about the world.

This layered construction can make social simulations harder to develop and test: a “vertical slice”³⁰ keeps getting deeper and deeper as development progresses. Exploring the hidden depths of a social simulation, however, can be a unique charm unlike the spatial exploration in other kinds of games. Some techniques for tackling this design and authoring challenge are discussed below.

5.4.3 Stories About Specific Systems, not Specific People

Since most content for social simulations is written for patterns of interaction, rather than specific characters, what arises perhaps unsurprisingly tends to be stories more about a particular social milieu than specific characters within it. In fact, social

³⁰In game production terminology, a vertical slice demo is one that includes all or nearly all the features of the game running for a small portion of it, such as a full-featured prototype of a single level.

simulation's focus on rules, large casts of characters (to best show off the system), and primarily social actions gives it much in common with the existing dramatic tradition of the comedy of manners:

The subject of comedy of manners is the way people behave, the manners they employ in a social context; the chief concerns of the characters are sex and money (and thus the interrelated topics of marriage, adultery and divorce)... They [the characters] are playing a game, perhaps, but in deadly earnest and for the highest of stakes; and, moreover, they must stick to the rules. These rules are society's unwritten laws regulating behaviour, the dictates of propriety which, though they may differ in detail from age to age and class to class, are always basic to the conduct of the characters... (Hirst, 1979)

This describes quite nicely the systems and dynamics of social simulation games: domain-specific schema, adherence to rules, and a tendency not to break out of the norms dictated by those rules (discussed further in the next section).³¹ The comedy of manners often includes one or more characters who violate these strict social norms, of course: we will discuss in the next section how this role is often performed in a social simulation by the player.

Another mode we might fruitfully compare social simulations to is ethnography. Scholars and authors dating back to Edith Wharton (Bentley, 1995) have noted the similarities between fiction writing and anthropological ethnography “as related forms for analyzing and exhibiting social life.” This approach puts more perspective on the power of the writer (or in our case, the designer encoding social rules) to either uphold or subvert social norms:

³¹The focus on sex and money seems at first a red herring, but consider that nearly every game mentioned in this chapter features romance as a significant aspect of the simulation (except, unfortunately, *Football Manager*).

Viewed in this way, manners are a form of active regulation, the installation of a social order deep within the body and personality of the subject. Similarly, the activity of writing about manners is a contingent—though not identical—process of supervision, providing as it does a natural-seeming account of the fashioning of the self as a rounded, social character. To recognize as much, though, still leaves open the question of the nature of that supervision, whether the portrait of manners in a novel is intended either to reinforce or to subvert the internalized “cosmology” that is the matrix for the controlling laws of decorum. Does writing about manners defend or undermine the hierarchies they serve? (p. 7)

The question of whether social simulations defend or undermine the social schema they encode has come up in the context of game design. *Redshirt* features a playable race of women: “the classic sexy green alien race—[who] often receive lewd Spacebook comments from sleazy men, even if they’ve listed their sexual preference as being interested only in women.” A player who had experienced such abuse in real life felt this aspect of the simulation was unnecessarily threatening—they interpreted the game as a normalization of this behavior, not a commentary on it—leading the designer to issue a public apology and clarification of her intent (qtd. in Sarkar, 2013). Reviewers of *Prom Week*, too, often complained of “feeling bad about myself” (Pearson, 2012) or “crossing [a] line” (Stephens, 2012) by being complicit with a shallow system of social manipulation, or being unsure how to perform a rejection of that system from within the simulation.

In part, reactions like these stem from our lack of a cultural understanding or critical vocabulary for processing playable simulations of human interaction, as much as a lack of design wisdom built up through generations of similar games. As more such games are created and released, we will doubtless develop more strategies for making,

playing and processing them.

Both ethnography and the comedy of manners share the characteristic of focusing more on patterns of behavior and typical (or stereotypical) characters rather than the specificity of character that most forms of fiction foreground, and are thus useful parallels placing the social simulation storygame within a larger context. While storygames are of course different artifacts with their own challenges, think again of our hypothetical *Fellowship* game. We might argue social simulation is a good match for this game by observing that, while Tolkien's work certainly contains specific, memorable characters, they are often used as stand-ins for his fictional cultures. The four hobbits are each variations on a theme of a simple pastoral life; the dwarf Gimli and elf Legolas each a literal representative of their race and its cultural concerns. The humans Aragorn and Boromir exist to together offer a contrast in how mere mortals respond to temptation. The way this set of characters interact with each other is closely concerned with their social values and expectations—and the moments when they question, defend, or transcend them. We might argue that Tolkien's story is therefore a better match for adaptation via social simulation than a fantasy story like Ursula K. Le Guin's *A Wizard of Earthsea*, much more concerned with a few solitary, sharply defined characters on personal journeys. This sort of critical design thinking is more difficult without an understanding of how an emerging mode of gaming like social simulation relates to broader cultural practice.

5.4.4 Breaking out of Ruts

Because released social simulations have tended to eschew long-term planning approaches in favor of characters that simply react to the current situation,³² characters can have a tendency to get stuck in ruts. Given a social situation and a set of rules and actions, characters will most likely be inclined to take actions that reinforce their circumstances: continuing to treat friends nicely, enemies harshly, and otherwise behave in ways that maintain the status quo their traits and associated rules dictate are desirable. Without any change to the state or behaviors, characters tend to drift to local maxima and minima, edge conditions where numerical relations hit their most extreme possible values.

All the systems surveyed earlier use techniques to prevent this from occurring:

1. In *Prom Week*, the player by default can only take actions with a positive volition (the character naturally wants to perform that action). Each time they perform such an action, however, they acquire Social Influence Points, which can be spent to unlock actions with a negative volition. This allows the player to occasionally force characters to act outside their desired behavior.
2. In *Blood & Laurels* the system provides the player socially appropriate actions, but also lets them occasionally choose inappropriate ones. This gives the player the option of being the agent of change. The game also frequently introduces new narrative situations that offer the player different opportunities to change the social state. Finally, when the state reaches an edge condition, the social practice model might open up a new action with drastic consequences on the

³²Why is this the case? I honestly don't know, but it's a observable fact that very few games with complex goal-planning characters have made it to release. The additional technical skills required to implement and design for a planner may be part of it. Michael Mateas suggests in a private correspondence that planning may be fundamentally at odds with player agency, which must either be limited or trigger a continual and computationally expensive replanning; accounting for player agency to begin with, as in search-based drama management (Weyhrauch, 1997), is even more computationally expensive.

social state, such as an action to assassinate someone for whom your affinity is approaching the lowest possible value.

3. *Redshirt* uses an element of randomness to make NPC decisions regarding romance less predictable. The player's actions are also unrestricted by social norms, again allowing them to be an active agent of change.

This is another function that narrative systems can provide to enhance a purely social simulation: to continuously alter the story context to prevent a static situation. The unfinished mid-80s story generation system Universe, for instance, included the notion of author-level goals such as “churning lovers,” introducing a new complication between romantic partners to keep them separated (Lebowitz, 1984). The goal in this case was to create stories modeled after soap opera plots that continuously evolved instead of moving towards conclusions, but even within traditionally bounded stories the notion of churning the social state to prevent stagnation is clearly useful.

Another solution to this, as seen in the examples above, is to have the player character be the one who churns the social state, by freeing them to take actions outside social norms. We have discussed how social simulations echo the tropes of the comedy of manners, but in this aspect they suggest another literary character type: the picaresque rogue, who moves between constrained social groups and is free to violate their conventions.

Even with all these solutions, the dominant state is still for a character to perform their given social role. This is another technical artifact of these systems that reinforces a procedural rhetoric in tune with stories about a social milieu, not necessarily

about heroic characters breaking out of that milieu. (Which is not to say such stories can not be told with social simulation; just that the procedures generally encourage an underlying state where such is not the case.)

5.5 Challenges with Social Simulation Games

Social simulation storygames are radically different from most existing storygames in their design, implementation, and evolution. Designers face a number of challenges and potential pitfalls. The remainder of this chapter contains a detailed exploration of these challenges along with approaches designers can take to successfully mitigate them. This wisdom, compiled from multiple projects, represents a major part of the design contribution of this chapter.

Specifically, the following sections will look at difficulties in three aspects of producing social simulation games:

- During **design**, designers will face questions of how to prototype a social simulation, decide what level of complexity is needed to achieve design goals, and how to make the system visible without overwhelming players.
- During **production**, authors will need techniques for managing rules, and for knowing how to write sufficient narrations and discover and fix bugs within a simulation space.
- Designers need to also ensure the **output** of a social simulation avoids the aesthetic issues arising from re-use of text, and is also neither boring nor offensive.

5.5.1 Prototyping and Design

Most existing storygame genres consist of discrete scenes that can be prototyped and played in relative isolation. A classic adventure game might be comprised of a series of rooms; a choice-based story as a network of nodes. The “vertical slice” approach to design mentioned above can be used to build up one of these rooms or nodes, or a small connected set, as a representative stand-in for the whole experience: a complete moment out of a story that will be comprised of a series of them. However, since one of the definitional qualities of social simulation storygames is that their charm comes from depth of continuous interaction and not a broad selection of designed moments, how can we prototype a vertical slice of such a game? How can designers get a feel for how a game is developing before the full set of schema, rules, and narrations is in place? Is there a way to sketch or prototype a social simulation?

Design by aggregation. One approach is to grow the social world slowly from a few small seeds, rather than architect its outlines first and fill in the details later. On *Prom Week*, the team did not take this approach: early on the full social schema was designed, and a full set of rules was then laboriously developed to describe how these aspects affected character volition. Only then was authoring begun in earnest to create narrations to describe all these possible states. As a result, through much of the testing process the system’s ability to narrate the state lagged behind the complexity of the simulation, preventing the game’s surface from revealing the complexities going on underneath. This meant a lot of design insights about the utility of certain aspects of

the schema came quite late in development.

A different approach might have been to start with a very simple social concept (for instance, the notion of friendship) and determine how to capture that through a minimal set of schema, rules, and actions. Playtesting could happen in a world where friendship is the only social factor considered. Once this concept was fully exercised, and modified as needed, additional complexity could be added: such as the notion of one character becoming temporarily annoyed with another.

There are two potential difficulties with this approach. First, each new “concept” added to the system requires considering how it interacts with every old concept, and possibly reconsidering earlier decisions. Being annoyed with someone may well be a state that should modify existing friendship rules. By contrast, when building a system from a plan, each aspect of it can be considered when building out a particular component. The ability of rules to be additive alleviates some of this, however: if a naive friendship rule increases a weighting by 5, and a new annoyance rule decreases it by 3, the only intersection point is the relative weightings.

Second, how do you know when you’re finished? Starting with a plan, you have at least a rough metric for when you are done: when you have written enough material to cover the state suggested by your social schema. However, growing a schema is potentially a never-ending process. This difficulty will be discussed in more detail in (5.5.4).

Design through user stories. A related approach is to begin the design process with a set of stories you’d like to be able to tell in your domain, and grow the set of rules and narration content one story at a time. Discussed later in the context of simulation fidelity (5.5.2), this story-first approach can be a useful way of only adding what is strictly necessary to the simulation. For instance, in a comic game about meeting the oddball parents of a significant other, a user story might be “It should be difficult to get the parents to like you.” This can then drive development: we clearly need to implement liking and disliking, we need traits or other features to distinguish how the parents’ behavior differs from the player character’s, and we need rules and systems in place such that it takes more than a single move to achieve this state.

Paper prototyping. Though social simulation seems computationally intensive, high-level mechanics can be roughed out through paper prototypes. A gamemaster can have a list of rules, and consult them when the player takes a social action. (This list can productively be simplified: for example, instead of a series of specific rules about annoyance, a single meta rule that being annoying makes people less likely to interact.) The gamemaster can explain the social state and, if necessary, what actions are available to the player at any given time. This can give a good sense of the space of actions that will be necessary, what rules are missing, which existing rules / schema elements are dominating or rarely used, and so on. Mixed analog/digital prototypes are also possible: a description of how we prototyped *Prom Week* this way can be found in McCoy et al. (2013).

5.5.2 Simulation Fidelity

How complex does any simulation need to be, and specifically, how complex does a social simulation in a storygame need to be?

The most obvious answer is “just complex enough to be interesting,” and there appears to be no lower bound on complexity when this is the metric. Conway’s Game of Life (1970) famously has only four rules, but enough emergent complexity to inspire decades of analysis and even entire conferences dedicated to studying it and other seemingly simple cellular automata.³³

We might also consider *Parable of the Polygons*, a 2014 project by Vi Hart and Nicky Case which uses an extremely simple social simulation to demonstrate how urban segregation can happen even with well-meaning actors. Characters in *Polygons* are simulated with just two qualities (*shape*, which can be square or triangle, and *mood*, which can be happy or unhappy), and a single rule: “I wanna move if less than 1/3 of my neighbors are like me.” In a series of interactive steps, the creators demonstrate how this gives rise over time to a pattern where the squares and triangles become segregated into zones with no outsiders as unhappy squares move at random to new locations. The outcome of the simulation is surprising—the rule suggests the shapes prefer diversity, not homogeneity—and the user is given expressive input in the various components, able to rearrange initial configurations of shapes and experiment with altering the parameters of the rule.

³³Including AUTOMATA, celebrating its 23rd year in 2017, and ACRI, the Conference on Cellular Automata for Research and Industry.

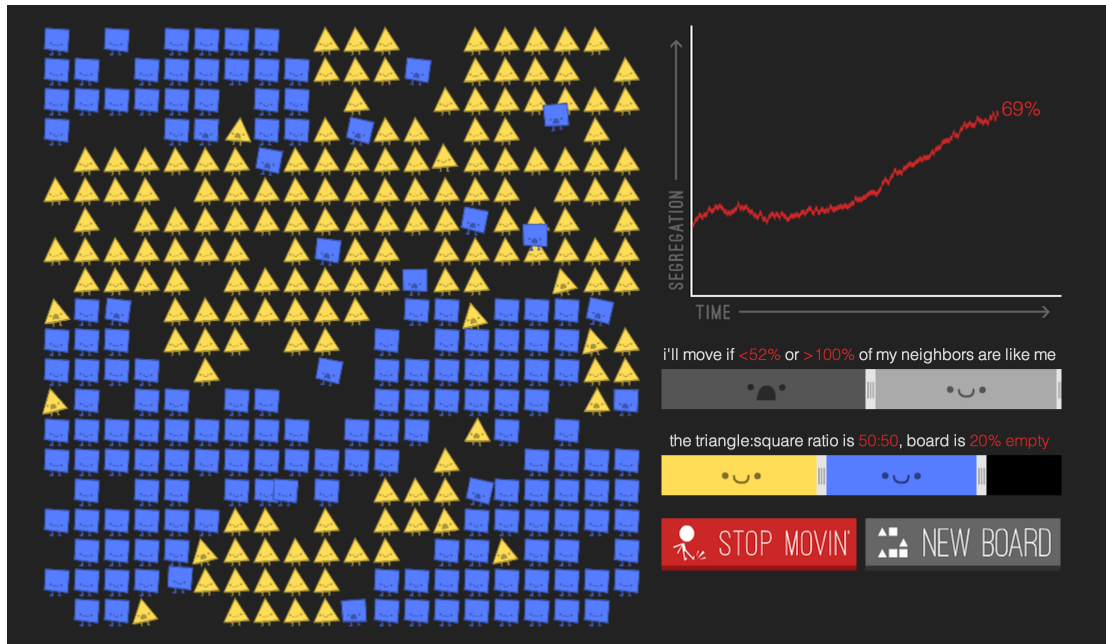


Figure 5.10: Screenshot from *Parable of the Polygons*.

Polygons does not have narrative except through implication, and while entertaining to play with, its implications can be exhausted in five to ten minutes. The three storygames discussed earlier in the chapter all feature narrative scaffolding, which signposts to the player an expected playtime by moving them through a series of scenes, with a limited number of moves available in each. These structures imply that the designers expected their social simulations to support gameplay enough for at least one traversal through this structure, and perhaps several. We can thus infer each designer expected players would interact with their game for significantly longer than with *Polygons*, and that the increased complexity of the social simulation was designed to support this. Adding narrative structure via frequent changing of scenes and shuffling of characters is one way of adding playtime, but in each case also acts as a way of showcasing different

aspects of the underlying simulation, which must be complex and interesting enough to support the entire traversal.

But it's also clear that playtime does not increase linearly with complexity. Versu features hundreds of elements in its social schema and *Prom Week* has thousands of rules, but neither of them are narratively structured so as to suggest playtimes hundreds or thousands of times longer than *Polygons*.

We might therefore make the following two assertions:

1. Increased social simulation complexity increases playtime, but
2. This increase does not scale linearly.

In Figure 5.11 we therefore arrive more logically at our initial gut impulse: a social simulation should be just complex enough to be interesting. That is, it should have just enough complexity to support a traversal through the narrative structure the designer intends to create. Additional complexity beyond this sweet spot trades increasing amounts of work for decreasing amounts of player enjoyment. So how does a designer find this sweet spot?

A social simulation is generally trying to make some aspect of real or hypothetical social interactions seem plausible: interactions between stereotypical high school characters, career-minded space cadets, ambitious schemers in ancient Rome, and so on. For the simulation to seem genuine, enough of the actions and behaviors associated with these milieus must be simulated so that the most plausible actions have a believable and interesting range of possible consequences.

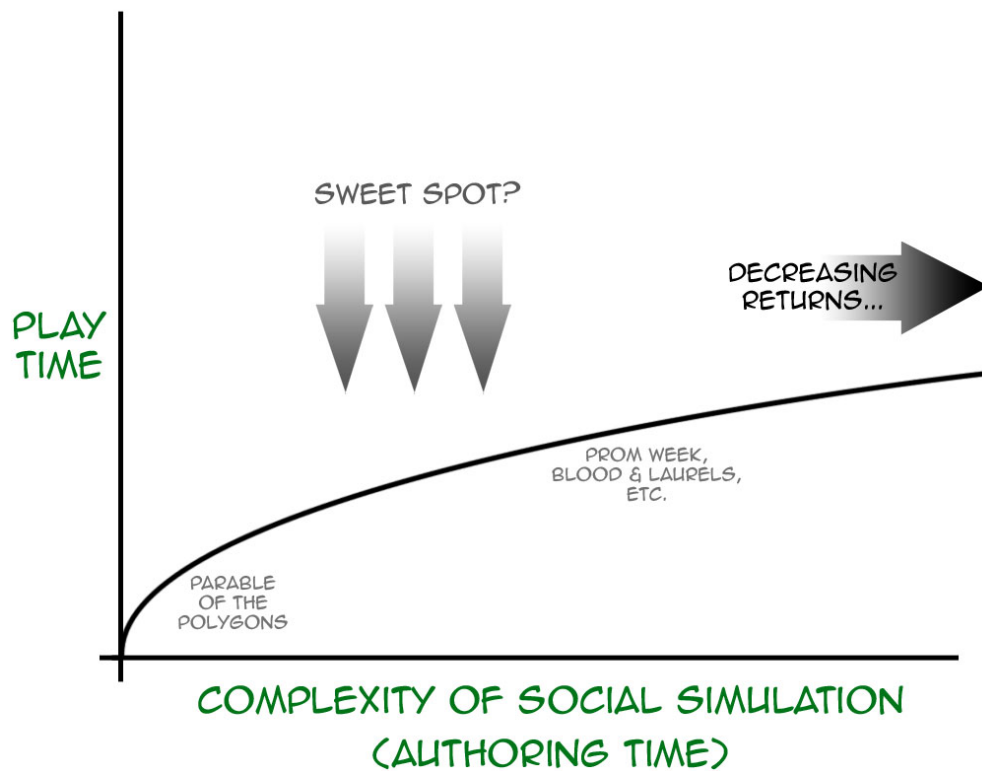


Figure 5.11: Evidence suggests social simulation play time does not increase linearly with the complexity of the simulation, or the work needed to author content to realize that complexity.

Many designers begin to define this range by analyzing non-interactive works in the target domain, and formally or informally coding them for patterns that seem representative of the space of behaviors, details, and actions that take place in the domain. The designers of *Dwarf Fortress*, for instance, write short stories set in a fantasy world like the one they wish to portray, and then review these stories looking for elements not yet part of the game’s simulated world.

In one of his [hand-authored stories] titled ‘Battlefield Lunch,’ [*Dwarf Fortress* co-designer] Zach details a range of activities surrounding the carrion left in the wake of a large battle. ...Based on this short episode... the Adams brothers discover several behaviors not yet simulated in *Dwarf Fortress*.

Among their notes are the following:

1. large scavengers eating the dead
2. entity hatred can manifest through acts of torture on those that can't defend themselves
3. wrestling: tripping people, becoming impaled on ground objects

By analyzing the narrative to extract its implicit tropes and conventions, the Adamses attempt to translate the rules of the storyworld into the algorithmic processes running *Dwarf Fortress*. The hope is that, by identifying and encoding patterns such as revenge, scavenging, and grappling, narrative events akin to those described in ‘Battlefield Lunch’ will propagate widely across the simulation. (Boluk and LeMieux, 2013)

Similarly, the *Crusader Kings II* team also studied historical events for the kinds of actions that princes, regents, and assassins took towards each other (Kaiser, 2013), while the *Prom Week* team analyzed classic movies and TV shows set in high school to devise an initial list of important social statuses, actions, and behaviors; the first draft of the simulation schema came from these notes (McCoy et al., 2013). While not all social simulations need to aspire to the level of detail of *Dwarf Fortress* (or even *Prom Week*), identifying the foundational units of a social milieu is a critical first step in a satisfying simulation of it.

Another question is how important social history is to a social simulation’s effectiveness. Some interactive story authors have argued in favor of abstracting specific historical events away into aggregate stats, to reduce complexity and minimize the amount of state that must be tracked and reasoned over (Fabulich, 2011; Kennedy, 2016). Anecdotally, the *Prom Week* team found that references to specific past events were one of the most satisfying and memorable “tricks” the system could perform: some

preliminary analysis work also suggests this as a quantitative result (Samuel et al., 2014). However, the code to enable any character to later describe an event using the correct tenses and pronouns was complex: this is discussed further in (6.1.2).

While there is no simple answer to the question of simulation fidelity, the prototype and design techniques described above suggest an initially simple design, layering in complexity into the system until it feels robust enough to tell the kinds of stories the designer envisions.

5.5.3 Making the System Visible

A frequent problem for games with complex simulations is presenting the player with enough information to understand and operate the system, but not so much that they are overwhelmed or confused. The tension between these two extremes is described by Wardrip-Fruin as the difference between the *SimCity* effect and the Tale-Spin effect (Wardrip-Fruin, 2009, p. 115, 299), referring to the successful city management simulator and a story generation system with complex internal logic but simplistic surface output which often does not expose the interesting internal processes.

Designers of complex systems are caught between two potential dangers. By not revealing enough about the internals of the system, they risk players misunderstanding or dismissing the extent and value of those internals, as in Tale-Spin. Emily Short, for instance, has described how players of Versu-driven games often do not understand that the text they're reading is dynamic, assuming everything they're seeing is hand-authored: the system only became visible when it did something wrong (Short, 2015b). On the other

hand, exposing too much information can make a game difficult to learn and operate. Critics have written about how games like *Dwarf Fortress* are frustrating because of the high cognitive load required to process and apply all the available information about their underlying simulations (Johnston, 2013).

Designers of storygames stumble especially on this problem, because of the perception, right or wrong, that ludic elements such as visible numbers detract from a narrative experience. Chris Crawford, for instance, has explicitly made a rejection of visible numbers a feature of his latest project: the Siboot site touts Parsimony as a fundamental value, saying that the player’s interactions with characters will need to rely on “intuition, not computation. . . You can never know enough to compute the best course; you must rely on your intuition.”³⁴ For *Faade*, Mateas and Stern have discussed how their decision to communicate all important game state through diegetic means such as facial expressions and dialog cues meant many players did not pick up on these details, leading to a less intentionally-driven play experience than they had originally envisioned (2005). One could argue that a game like *Fallen London* (discussed in 3.2.4.1) makes its play experience too mechanical by exposing so many numeric stats about the fiction.

Beta testing can again be helpful, especially if designers can question players afterwards about their decisions. If many players are taking non-optimal actions because of misunderstanding or being unaware of critical aspects of the simulation, the designers may need to find ways to make those aspects more visible.

³⁴<http://www.siboot.org>, accessed May 6 2017.

5.5.4 Authoring and Scoping

Once you’ve successfully prototyped a social simulation game, how do you know when to stop? While the approaches described above of designing by aggregation and designing around user stories can be helpful for knowing how many rules you need to author, a separate question is how much narration is required to successfully explain the state of that simulation. Again, this is easier when authoring for a choice structure: the amount of content within each lexia might vary, but you know from the beginning that each only needs enough content to describe itself and the following choices. Authoring for social simulations does not provide these familiar milestones in the same way.

At a minimum, there must be narration to explain every possible social state change. Assuming most social actions can potentially be bidirectional (either raising or lowering a number, or starting and stopping a state), that means twice the number of possible social actions. For a system that also has the notion of an action being accepted or rejected, as in *Prom Week*, that multiplies the minimum number of scenes by two again. *Prom Week* required at a minimum 24 scenes to narrate changes to its three network values and three relationship values ($6 \times 2 \times 2$), with labels like *Generic Start Dating Reject* or *Generic End Friendship Accept*.

For some systems, this bare minimum may be enough. *Redshirt*’s device of explaining all social actions through a social network means a report such as “Jo likes your status!” is all that’s required to narrate the social action. But a system telling a story through more traditional narration may require character dialogue or descriptive

text to narrate a social state change. Ideally, that text should be as contextual as possible, or at the very least it should vary each time the player sees it (discussed later in 5.5.7).

One approach to this contextualization is to subdivide each atomic action based on narratively distinct ways a situation might play out. For instance, a fantasy game might want to describe an elf expressing anger quite differently from a dwarf doing the same: this subdivides narration of the Express Anger action into one version for elves and one for dwarves. A Start Friendship social action might be subdivided into a version for characters meeting for the first time, and another for characters patching up an old argument.

But determining how and where to make these subdivisions, and how many a particular game needs, is its own problem. The more subdivisions, the more ways a lexia can be specific, as human authors know more details about the situation that can be inserted into the text (following the writing advice that adding interesting specific details leads to better stories). But the more lexia we make, the more we increase our authoring burden: and the more specific a lexia, the less likely it will ever appear in a particular traversal. As the number of specific lexia increases, more and more authoring time has been spent on content that individual players seldom see: as this becomes more extreme, most authored content will never be seen by any player.

With these challenges in mind, how do we best scope authoring social simulation narrations?

Even subdivisions along heterogeneous narrative axes. Authors look for aspects of the simulation that might split the possibility space roughly in half. For instance, if there are a roughly equivalent number of good and evil characters, a separate version of certain actions for each might be a useful subdivision. It can be difficult, of course, to estimate how likely certain scenarios are to occur in advance, but this can be a good starting point.

Author for exceptions and edge cases. Instead of painting with broad strokes, think instead of interesting variations. For instance, an interesting breakup scene might be one in which the recipient is highly confident in the relationship and doesn't see the action coming. A Heroic Rescue action might be interestingly different in the unlikely event it's performed by a villain. This can produce interesting dramatic moments and make the system seem very smart. However, the danger is that these scenes are so specific they only rarely (or never) come up in actual play.

Author proactively based on ideal user stories. Just as user stories can help during prototyping to determine what kinds of social actions are needed to support the kinds of stories you want to tell, these same user stories can be used to help subdivide narrations. Many of the unique situations in a game's user stories might suggest subdivisions to help make narration of those scenarios more distinct. This is limiting in that it's focused on stories you've already thought of, not the massive emergent space your system enables, but can help ensure you're meeting your baseline for the kind of stories you want your system to tell.

Author reactively based on non-optimal test playthroughs. Another approach is to collect from testers a set of stories that fell flat, and author new narrations to cover those specific situations. Frequent playtesting during design can help catch scenarios your narration is not yet adequately covering, or that designers hadn't previously considered.

Prioritize subdivisions based on frequency of actions. Also a factor is how often individual actions are employed by players or simulated characters. Focusing authoring effort on creating variations of the most frequently used actions is more useful than spending time on rarer actions, which can get by with fewer unique narrations.

In practice, a combination of these techniques is likely to be most useful for any given project. Testing (detailed in the next section) can also help with this process. A random tester, for instance, might help determine which lexia are most or least commonly used. Overused lexia can perhaps be split into more specific variants: Underused lexia might want to be made more generalizable by removing a precondition.

Designers can use the rate of repetition across an average tester traversal as a rough guidepost for when their pool of narrations is rich enough. If the designer hopes the experience will last half an hour, yet towards the end of this time in an average play session (or an automated, random playthrough) the player is often seeing lexia they've already encountered, this is a sign either more content should be authored or the optimal play time should be reconsidered.

5.5.5 Testing

Three techniques from software development can be especially useful in gaging the quality of a social simulation in progress.

Random Testing. Described previously in (3.4.1.2), this technique involves simulating thousands of playthroughs with moves made either randomly or according to some heuristic for player behavior. With its huge possibility space, a random tester can be a helpful way of exploring the boundaries of a social simulation. As with its use verifying sculptural fiction, however, some metric is needed to detect broken or non-optimal states. Broken states should be straightforward to define (a numeric value should never exceed its maximum). Non-optimal states might be defined on a per-character or per-scenario basis. For instance, for a given scenario meant to be challenging, a criteria might be that no playthrough should be able to complete it in a single move. Be sure to distinguish states that are unlikely but possible (and potentially narratively interesting) from states that should actively be pruned out of your possibility space.

Cohesion (Integration) Testing. Having human testers play and report back on any narratively or socially unbelievable scenes is another lens on how well a social simulation is performing. Each of these reports provides a direct pathway to tweak something in the simulation to make character behavior more believable. This is a scatter-shot approach that can feel daunting in the large possibility space of an extensive social simulation, but can also help identify problem areas that other testing regimes aren't discovering.

Regression Testing. Maintaining a large suite of expected inputs and outputs, and adding to it as bugs are fixed and new behavior is introduced, can be a useful technique, although it faces several challenges in the context of a simulation. First, iteration often takes the form of slowly tuning rules until they “feel right”: weighted rules, for instance, might have their values tweaked during playtesting to accommodate the relative importance of various social factors. This tuning, however, can cause a wide variety of changes, subtle and more obvious, across a suite of regression tests: even if many of the new responses are still valid, the tests now all need to be updated. (That is, there is a large space of both correct and incorrect responses within a social simulation, as opposed to many other aspects of code that can be tested where there is only one right or wrong response.) One approach to this is to have “fuzzy” regression tests that look for ranges or patterns rather than distinct values. Another is to only test situations that must work for the game to be completable.

5.5.6 Rule Sorting, Precedence and Weighting

When creating rules for a social simulation, how should they be structured, and how should they take precedence over each other?

Creating a large number of production rules with no structure or hierarchy has been derided by some AI specialists as a “bag of rules” approach to authoring: “the view that expert knowledge can be encoded as a uniform, weakly-structured set of if/then associations is found to be wanting” (Clancey, 1983). Social simulations have used various strategies to avoid this. CiF’s creators, for instance, added a concept called

microtheories to cluster a set of rules together with the same precondition, “an authoring strategy that helps tame the complexity...for example, `relationship(friends,x,y)` is the definition of the Friends microtheory. ...The rule set then provides a general understanding of what it means to be friends.” (Mateas and McCoy, 2014)

While I agree complexity management is a problem, I do not believe a hierarchical approach such as microtheories is the best solution, for two major reasons. First, any rule in a social simulation is inherently a connection between two or more different concepts in that simulation’s schema: for instance, the rule “Being cheerful makes one more inclined to start friendships” connects an ephemeral status of *cheerful* and a more stable relationship of *friendship*. Should one, then, place this rule in the “cheerfulness” or “friendship” hierarchy? An argument based on computer-facing preconditions weakens the human-facing justification for this technique (that it allows one a “general understanding” of a concept) but clearly the rule above connects to both cheerfulness and friendship equally. Rules like this are designed to connect potentially any two concepts together anywhere in the simulation, and are thus inherently anti-hierarchical. Second, such hierarchicalization implies a top-down design that limits author-time flexibility. Organizing rule knowledge into frameworks “[can] make it easier for an expert to organize his thoughts” but “this meta-level analysis may impose an extra burden by turning the expert into a taxonomist of his own knowledge—a task that may require considerable assistance” (Clancey, 1983). Given that most game designers are not also sociologists, the term “expert” may not apply in the same sense it does for specialist interviews collecting expert system knowledge. An author’s (or artist’s) flexibility to add and

remove, revise, iterate, and explore within an evolving space of social rules, in my view, should trump structures that require more limiting hierarchical planning in advance.

However, this still leaves us with the problems of organization and complexity management we began with. A clear solution is to support both author-time and run-time tools for creating rule groupings ad-hoc and as needed. If an author can easily search for, view, and edit any rules that mention cheerfulness or friendship, the need for a baked-in structure is clearly reduced. This was the approach taken by the Ensemble project (6.2), which does not natively include a concept of microtheories. A more complex but conceptually similar technical problem is in reducing the computational burden at run-time by eliminating rules from consideration that don't apply in a given circumstance. A full technical solution is outside the scope of this dissertation, but approaches such as the rete algorithm (Doorenbos, 1995) have previously been devised to solve this exact problem.

A second difficulty is that of precedence. We earlier (5.4.1) observed that a set of rules can have two broad weighting strategies: *aggregate*, where each rule is weighted equally and thus the largest number of matching rules takes precedence; or *weighted*, where each rule has a numeric weight and the highest total weight takes precedence. Aggregate rules are the simplest system, but make the unrealistic assumption that all possible social considerations are equally important: i.e. when deciding whether to accept a marriage proposal, the fact that the proposer is currently cheating on the proposee is just as important as the proposer's current mood of mild annoyance for some unrelated reason.

Given that weighted rules seem more realistic, the problem then becomes how to scale weights effectively. What level of precision do weights need? What if an author discovers halfway through rule creation that more precision is needed: do all previously written rules need to be re-weighted? How do multiple rule authors agree on a scale for assigning weights?

One solution is to use *weight labels*, rather than direct numeric values, and use *as few labels as possible* to accurately reflect reasoning in a particular social space. The simplest version of this is two weight labels, *Normal* and *Important*.³⁵ These are assigned a specific numeric value in exactly one place, to be tuned later on if necessary. More labels and gradations can be added, but only if necessary for realistic behavior.

A connected problem is preserving an author's rationalization for giving a rule a particular weight. What if one author lowers the weight of a rule to fix an issue with a certain scenario, but this breaks another scenario, causing another (or the same) author to later change the weight again? This is another reason to use weight labels and use as few as possible: ideally, it should be clear to all authors which level of importance applies to any particular rule; or at least disagreements should happen as infrequently as possible.³⁶

³⁵The urge to create "trump card" weight labels should be resisted, because this undercuts precisely the capacity of a social simulation to produce emergent surprises. For instance, it may well be that a character would agree to a marriage proposal from someone they knew to be dating another person, for all kinds of reasons: an inclination to polyamory, a deep and desperate love, a practical need for marriage (such as acquiring a green card) and so on.

³⁶Social rules are of course inherently subjective, so this problem can never be entirely designed away. Different authors (or the same author on different days) might weight the relative importance of various social factors quite differently based on their cultural background, recent media consumption, social attitudes, or mood, even with only a few levels of gradation. For instance, someone who's been cheated on before might rate the importance of such a consideration higher than someone who hasn't.

5.5.7 Reuse of Content

In a storygame driven by directed choice graphs, the player moves forward from node to node through a series of hand-authored lexia. The fact that the story is merely fictional, not simulated, can be disguised by the fact that no piece of writing is seen more than once.

However, in a simulated story space described through written lexia, the player may re-encounter text they’ve already read during a single traversal. Curiously, while repetition is unremarkable in prerecorded audio sound effects or visual game elements such as background art or sprites, it grates when reading text: apparently, we don’t like to be told the same story twice. In the context of a simulation, repetition of text can make a system seem flat and mechanical. Even the original *Adventure* would only print room description text once by default, cutting back to a short summary afterwards.³⁷

While procedural generation in contexts such as creatures, scenery and sound effects has made significant advances, our ability to generate text for a changing situation is still quite limited. Designers of simulation-based storygames are therefore put in the difficult position of needing to use a limited supply of hand-authored lexia to describe a changing simulated story world. What are some solutions?

Playtime limited by content. Some systems connect running low on fresh content with mechanics to end the game, such as the story-centric board game *Betrayal at House*

³⁷While in the original game the shorter text was probably used to save teletype printer paper, the technique was preserved in text games long after this practical matter was no longer an issue: the “brief” mode emulating this technique is still the default for major interactive fiction languages.

on the Hill (2004). Exploration of the physical space of a *Betrayal* story is limited by the number of haunted house tiles shipped with the game, and the climax is triggered with increasing likelihood as the supply of narratively-charged Omen cards runs low.³⁸

Remove already-seen content from consideration. Many interactive story systems will remove a piece of content once it has been seen by the player, often in addition to the technique above (in a board game context, discarding cards rather than reshuffling them). This is often used to accelerate the process described above.

Summarize lexia that have already been seen. In addition to room descriptions, some parser interactive fiction uses this technique to replace quoted dialogue with a shorter summary, such as “Bob tells you again about the restaurant”: this drier version can feel much less mechanical than a character repeating an identical piece of quoted dialogue (Dyer, 2011, p. 319).

Contextual customization of available lexia. Parser IF reuses system messages by customizing them to a particular situation, sometimes in quite complex ways, allowing the same underlying message to display as *Roger removes the sword from the sheath.* or *You removed your blade from it.*³⁹ Especially for straightforward messages, use of templating can go a long way towards making lexia that seem generative. More advanced

³⁸The game’s central twist is that “each game ends differently” via a booklet with special lexia and rules for a number of different endings: this approach is achieved by the brute force solution of providing more endings than sessions of the game that most owners will ever play. Only hardcore players will likely enjoy playing the same “Haunt” more than once.

³⁹The language Inform 7 has had robust capabilities for this since 2014; see <http://inform7.com/news/2014/05/07/new-2014-build-of-inform-now-out/>

techniques include the use of templates that can in turn generate complex descriptions of the state (see *Prom Week*’s speaker-specific summaries of past events, described in 6.1.2), or nested expansion grammars that can produce surprising results from a simple starting seed (Ryan et al., 2016; Compton et al., 2015). At its extreme end, this technique turns into pure natural language generation.

Use of existing or non-linguistic frameworks to convey simulated aspects.

Redshirt uses the existing framework of social media to communicate changes to the simulation, which removes the cultural expectation that each text be fresh and unique. Versu can use changing character portraits, rather than prose, to convey a character’s current mood. This technique is especially suited for bits of the social state that change frequently and need to be often communicated to the player. Another variant, using a non-linguistic symbolic language, can be seen in the Chris Crawford projects described in (5.3.4.3).

5.5.8 The Bad Parts of Emergent Surprise

Nothing guarantees emergent behavior will be either interesting or appropriate. Kate Compton has coined the term “10,000 Bowls of Oatmeal” for the danger that procedurally generated content can be mathematically unique but perceptually bland (2016). Agent-based simulations can likewise be harder to debug and risk producing boring results, which is especially problematic in a narrative context.

Far worse, generated content can be inadvertently offensive or even dangerous.

A Twitter bot I wrote for generating random street scenes from *Fallen London*⁴⁰ featured one template where a person accompanies another to a location, provoking a response from an onlooker. During testing, this template was once realized as: “A gentleman leads a young girl into an alley. You smile approvingly.” The generation space and list of characters and locations was broad enough that it contained unforeseen possibilities with ugly implications.

Other botmakers have made similar missteps with more serious consequences: in 2015, a Twitter bot using similar combinatorial techniques made a post construed as a death threat, leading its maker to be questioned by the police (Hill, 2015). And when an algorithm is learning or absorbing some of its content from the public, not all of whom may be acting in good faith, the problem can get much worse, as Microsoft discovered in 2016 when its Tay chatbot, designed to learn from its users, was taught racist, sexist, and abusive behaviors by users, and was shut down within 24 hours of launch (Greene, 2016).

Botmaker Leonard Richardson compares making a bot (or any storytelling algorithm) to being a ventriloquist: “Society allows a ventriloquist a certain amount of license to say things via the dummy that they wouldn’t say as themselves . . . But you can’t say absolutely anything and expect ‘That wasn’t me, it was the dummy!’ to get you out of trouble” (2013). Other botmakers have described bots as:

. . . semi-autonomous: they exhibit behavior that is partially a function of the intentions that a programmer builds into them and partially a function of algorithms and machine learning abilities that respond to a plenitude of inputs. . . . Who is responsible for the output and actions of bots, both

⁴⁰@SeenAtTheBazaar

ethically and legally? How does semi-autonomy create ethical constraints that limit the maker of a bot? (Woolley et al., 2016)

Semi-autonomy and ceding narrative control to algorithms adds an element of risk to creating social simulation games. While it may be impossible to eliminate the possibility that code will produce unacceptable output, we can consider two sets of strategies for minimizing the possibility of such a system being either offensive or boring.

Consider what text is appropriate to generate. A first step is to consider the scope of what should and shouldn't be algorithmic in the first place. Meg Jayanth, co-creator of the global-spanning adventure game *80 Days*, has spoken about early design questions of whether any text in the game should be procedural: the ultimate decision was that very little should be, because the story demanded a constant cultural specificity that would impossible (or offensive) to reduce to a generic system (2016). Appropriateness can be temporal as well as topical: David Lublin, whose Twitter bots remix TV news images and close captions, manually disables the captions on days tragic news stories break, to eliminate the possibility of an accidental juxtaposition of word and image being in poor taste (Collins, 2016).

Authoring tools to understand the generation space. The more quickly and easily designers can see a wide range of possible outputs, the more likely problematic situations will be noticed and corrected before release. Tracery (Compton et al., 2015), a tool for writing expansion grammars, constantly shows dozens of examples of generated output at author-time: this is far preferable to only seeing examples of generation one

at a time within a game.

Think critically and filter foreseeable bad outputs. In situations where an algorithm might use language from any external source, or recombine language from any source, many botmakers advise re-generating any output containing words from a filter list of known offensive terms as a baseline (Daly, 2016). Since most generated spaces are quite large, even words that might only potentially be offensive (or could appear in combination with other words to be offensive) can safely be excluded (Compton, 2016). Clearly, not all offensive output can be caught this way: Allison Parish notes that “when language is used for violence, it’s always relational, interpersonal, tactical, contextual—it doesn’t necessarily leave a lexical trace.” Parish advises that a solution to minimizing offensive generated language cannot be purely technical but must likewise be relational, interpersonal, tactical, and contextual.⁴¹ But a filter can act as a first pass, despite lacking any further context: Gillian Smith has noted that this approach is similar to the way some humans approach “political correctness,” understanding they should avoid certain terms even if they don’t know the full social or historical significance of doing so (2016).

More generally, eliminating generative subject matter that might potentially be problematic in unscripted interactions can also help. This is a reason why many games with player-made avatars (and the final version of my street scene bot) do not include children. However, designers should take care not to simply remove any aspects

⁴¹<https://twitter.com/aparrish/status/713024948147765250>

from their game that might be potentially controversial, such as race.⁴² These elements simply require thoughtful consideration during design, as do any other elements of a game.

Differentiate generated content from hand-authored content. Generated or templated prose, when done well, can seem so natural that the seams vanish. While this can be a desirable effect, reducing the visibility of this aspect of the system can lower the player’s ability to understand and appreciate what the simulation is doing. Many of *Prom Week*’s carefully templated scenes were unremarkable for players to whom they seemed handwritten; the creators of *Versu* have reported similar problems with audiences not realizing the complexity of their system (Short, 2015b). For *Ice-Bound* (described in chapter 4), we explicitly highlighted templated text, and also allowed players to tap this text and see why that particular variation was chosen. This visibility might seem mimesis-breaking, but offers players a window into the workings of the system. In addition, seeing content that’s offensive but clearly generated can be less troubling than seeing such content and believing it written by a human (Kazemi, qtd. in Jeong, 2016).

Allow reporting offensive content. Both during testing and release of a game with generated content, a clear and easy way for players to report something they found offensive can both minimize the damage of potentially improper content and

⁴²Among other reasons, representation of difference is important, especially in text: audiences will tend to assume gender, racial, and cultural norms unless the author explicitly specifies otherwise. Jayanth (2016) discusses attempting to subtly include characters of a variety of races and sexualities in *80 Days*, but when readers did not pick up on this it was as if she had deliberately omitted them. In the final version of the game she favored prose that might in other situations have been too on-the-nose, but was worth it to ensure a broader spectrum of humanity was being represented in the game.

help designers refine their systems to remove it. Both reporting that requires human intervention (such as triggering an email to the designer) and an ability for a player to reject certain types of content from appearing again in their game can be useful techniques.

Allow players to rapidly escape unideal situations. Since generation is cheap, try to make sure it's easy for players to move on to something else if they find uninteresting or troubling content. Quick-moving conversations are better for generation than a character (or landscape) the player would be stuck with for the whole game.

5.6 Futures for Social Simulation

Social simulation has driven three major critically acclaimed storygames to date, and been a significant component of games in other fields, from dating sims to sports games and empire-building strategy games. Stories are about people, and the fact that most game stories and characters are not playable—are fictional, rather than simulated—will continue to be a stumbling block to the maturation of games as a storytelling medium. Social simulation, though undoubtedly more complicated both to implement and design, offers a promising route for increasing the complexity of characters and thus the maturity of stories in gaming.

It's worth noting that the vision of social simulation outlined in this chapter is certainly not the *only* possible vision for social simulation. While the patterns I have observed have tended to hold true across released games so far—what we might call “early

period” social simulation storygames—they are not the only patterns that might develop. In particular, combining social simulation with other kinds of gameplay (as discussed in Appendix A on Project Yarn) produces necessarily different dynamics and design patterns. It’s quite possible to imagine very different kinds of social simulation games based around active planning by characters pursuing future goals, which would presumably produce very different kinds of experiences than ones in which most characters are content to sit around passively performing their role in the social state. Until such time as we are able to play these games, however, we can only describe the patterns that have emerged so far, which this chapter has done.

In this chapter, I have outlined several contributions to the project of social simulation. First, I have studied the mechanics and dynamics of three major social simulation storygames, and used this analysis to begin characterizing this new design space. This has led to new insights about the strengths and weaknesses of these games, how they fit into large storytelling traditions, and specific design wisdom for game makers interested in exploring this new branch of playable media. In the next chapter, more insights into authoring specifically for social simulations will be offered. In the next chapter I will also describe how I have helped reimplement the CiF social simulation engine into a new open-source engine, Ensemble, and created a second-generation design tool for creating rules for this system. The release of this engine and tool will be a significant step forward for lowering the barrier of entry to creating new social simulation storygames.

Better understanding and more easily creating such games moves us closer to

the vision with which we opened this chapter, of narrative-focused gameplay within a familiar fantasy setting. In our hypothetical scene (strategic downtime with Tolkien’s Fellowship), every action the player took had social significance, and understanding and navigating that social space had serious consequences for other, more traditional components of gameplay later on. The notion of “skipping” this cutscene would be as nonsensical as skipping the combat sequences in a first person shooter today: the story *is* the gameplay.

There are, however, still many steps that need to be taken to arrive at this future. Several key factors will contribute to the success of future social simulation storygames.

Accessible Tools. Currently there exists no off-the-shelf tool for incorporating social simulation into a game, at any scale. The release of the Ensemble Engine and Tool (discussed in the next chapter) will hopefully be a step towards addressing this. Especially within the indie world, the release of free and easy-to-use tools has in the past driven massive innovation in the kind and sophistication of storygames being made. Such revolutionary tools for storygames specifically include Inform (1994) and Inform 7 (2005), Ren’Py (2004), Twine (2009), and Tracery (2014), and in the larger games world RPGMaker (1992) and Unity (2005). Especially promising is the ability of these tools to work together: to that end the Ensemble team hopes to provide plug-ins both to use Ensemble characters within Twine, and to leverage Tracery’s expansion grammar tech and tools to help Ensemble characters express themselves in natural language dialogue.

We hope Ensemble will empower a much broader range of creators to start exploring the possibilities of social simulation storygames.

Shared Content Libraries. Social simulation lacks an ecosystem of bootstrapped content enjoyed by other gamemaking tools, such as the thousands of Unity-compatible 3D objects available for purchase online, or vast libraries of sound effects. Might we imagine some equivalent for social rules, the hand-creation of which takes up a large percentage of the time spent authoring a social simulation?

I'm skeptical that off-the-shelf libraries will ever exist in quite the same way, for the straightforward reason that in social simulation the library *is* the game: that is, each rule says something deliberate and intentional about the characters and their relationship to the world, and that process is difficult to automate. Good rules entangle the components of a social schema: additional components can't just be dropped in without an extensive pass of connecting them to all existing concepts, and modifying those old concepts for new aspects of the social simulation. Furthermore, social rules are subjective in a deeper way than objects and sound effects: two writers might make different playable stories using the same logic for flammable objects, but each of them is likely to have their own ideas about the nature of love.

A more promising approach, perhaps, is subtractive authoring: giving people an extensive library of rules they can aggressively prune, to focus in on (and modify) what they want to say about how their characters function. A rich library of annotated code and worked examples may also help, along with higher-level tools to quickly visualize

and understand how a social schema fits together. But the act of rule authoring for social simulation is inevitably the act of authorship: writers must become rule authors for social simulations to tell stories.

Integration with other game mechanics. In 2014, I worked with several collaborators at UC Santa Cruz and Microsoft Studios to explore integrating social simulation into a facsimile of an existing AAA game. Project Yarn (described in Appendix A) showed potential that social simulation could integrate with other game mechanics in a way that strengthened both systems, opening up a new dimension of both player input and system expression. While *Redshirt* does some work connecting social simulation to other game mechanics, most social simulation games to date drive games largely about social interaction. A promising future is seeing richer social components added to games that aren't primarily about social interaction in isolation.

More Proofs by Existence. While several social simulation games have achieved critical acclaim, there has not yet been a game in this genre that has had mainstream commercial success or widespread popular acclaim. This will remain a key barrier to finding the capital and will to explore such games further, both for industry and professional indie developers. As various supporting technologies such as speech-to-text and text-to-speech, natural language generation, face-tracking expression recognition and generative animation continue to mature, however, interest is growing in adding more dynamic and responsive characters to games.

Technical Evolutions. Work continues pushing forward social simulation technology to be more sophisticated. *Prom Week* co-creators Mike Treanor and Josh McCoy are working to extend the Ensemble engine to add Versu-inspired social practices (2016). Agent-driven storytelling platforms remain an active line of work in academia, although these systems still struggle to be used in mainstream games.

In the short term I believe the promise of social simulation may actually be not in finding further complexity, but in striving for elegant simplicity. The systems in the games discussed in this chapter remain complex, confusing, and inaccessible compared to the dominant paradigms for storygame authorship, beyond both the comfort zone for indies and the risk level for industry. The most promising future for social simulation is one where a system becomes streamlined and straightforward enough that a designer can use it to make a breakout hit. What is the simplest social simulation engine that still remains compelling, surprising, and playful? What does a minimalist social simulation storygame look like? To me, this remains social simulation's most fascinating unanswered question.

Chapter 6

Social Simulation Case Study

I discussed the game *Prom Week* in (5.3.1); as mentioned there it has also been discussed in several academic papers (McCoy et al., 2013, 2014) as well as extensively in the dissertations of my collaborators (McCoy, 2012; Treanor, 2013; Samuel, 2016). While this work has thoroughly explained the game, its design philosophy, and its technical implementation, one area that has not received as much attention is the process of authoring content for *Prom Week*, and by extension social simulations generally. I have established (5.4.1) that social simulations require a radically different kind of authoring than other kinds of interactive stories. It seems worthwhile to take a detailed look at how this authoring is achieved, and how we might better enable it.

In the first part of this chapter, I will describe in depth the challenges and process of authoring for *Prom Week* and its social simulation engine *Comme il Faut* (CiF). In the second part, I will discuss how some of these challenges led to the Ensemble project, a reimplementaion of CiF with an accompanying second-generation design tool

structured to make authoring social simulation content easier.

6.1 Authoring for *Prom Week*

6.1.1 Overview

During the development of *Prom Week*, I served as lead writer, overseeing a team that at times included as many as eight people¹ (a fairly large number of writers for an academic game). Together, we created over eight hundred templated scenes that CiF could use to narrate the game’s social state changes, as well as a handful of other content (detailed in 6.1.3; some statistics on *Prom Week*’s content can be seen in Table 6.1). First, however, I’d like to discuss templated scenes (or *instantiations*, as they were known on this project) as the primary unit of content in the game.

Authoring for a social simulation like *Prom Week* addresses the second part of the two technical challenges with this style of game, discussed in (5.5.4): narrating changes to a complex system in an interesting and believable way. I will define the problem more fully and then explain how we attempted to address it.

As a social simulation, *Prom Week*’s lexia have a different job than those in other kinds of interactive story. Rather than narrate fragments of a consistent narrative, these lexia might be used to construct all manner of narratives that exist within the

¹Every member of the core team contributed some amount of writing for the game. During the summer of 2011, we also worked with a team of undergraduate writers, a few of whom continued working with us past the end of summer, and we also worked with a few other graduate students creating content at various points during the project. I continued writing content throughout this process and through the game’s final release. The writing team was Zane Mariano, Corey DiMiceli, Duncan Bowsman, Garin Kessler, Alexander Baker, Daniel Cetina, Lauren Scott and Alexei Othenin-Girard.

Prom Week Statistics	
10	campaigns
18	characters
3	relationship types (reciprocal binary)
3	networks (directed numeric)
5	gradations of judgment on numeric networks (very low, low, etc)
12	unique categories of intentable action (relationships * 2 + networks * 2)
39	unique actions [social moves] (Woo, Ask Out, etc.)
34	statuses (21 undirected, 13 directed)
6	status categories (feeling bad, etc)
43	traits (undirected) (+1 gender)
3	invisible traits (wears a hat, etc.)
10	trait categories
26	CKB items
11	CKB truth values (gross, romantic, etc)
4	CKB subjective labels (wants, has, etc)
13	SFDB labels (cool, lame, etc)
7	mix-in types (buddy, pejorative, etc)
834	instantiations
5,609	lines of dialogue in instantiations
6,253	uses of mix-ins, about 1.1 per line)
12,432	choreography commands, about 2.2 per line
5,476	predicates (conditions and changes): 1,722 from instantiations, 3,190 from microtheories and triggers, the balance from SSUs and endings)
145	microtheories, with 3,044 influence rules
226	text message status updates
105	backstory trait definitions (avg 5.8 per character)
81	backstory history events
491	backstory network values (Monica has 66 romance toward Buzz) 27/char, or 9/char for each network)
51	backstory relationship values (Monica is dating Buzz) [3 per char, or 1/char/rel type; on avg, each char starts with one friend, enemy, and date]
26	backstory cultural knowledge base truth states (samurai swords are cool)
366	backstory cultural knowledge base subjective states (Gunter has samurai sword) [20/char]

Table 6.1: Statistics for *Prom Week*.

boundaries of the simulation, including narratives not thought of in advance by an author.

The fundamental action that must be narrated in a social simulation is a change to the social state. As actions happen, they alter some aspect of that state. Two people introduced are now acquaintances; a person might perform an action that makes someone angry at them, or share an experience that makes them both feel closer. To perform believably, simulated characters must have some idea of what potential actions in the social space are likely to result in what potential changes. In CiF, we set certain parts of the simulation “actionable,” meaning characters can have a volition to change that part of the state as it applies to themselves.

Since characters can accept or reject a potential social move, as a starting point for scene authoring we need a minimum of four actions for each actionable schema type: an accept and reject for both attempting to begin (or increase) a state change, and another for attempting to stop (or decrease) it. Note that because an action and reaction are indivisible, each of the above is a distinct unit. (This is in contrast to systems like Versu, discussed in (5.3.2), that split actions and reactions into separate units.)

Once we have this bare minimum (four scenes for each actionable type in our social schema), the system can provide a generic narration of any possible change to the social state. The specifics of a narration might vary significantly from one project to the next. For us, an instantiation was a complex XML definition specifying a short scene of around four to ten lines of dialogue between the initiator of the action and the responder (and sometimes a third party), each tagged with animation markers and other

supplementary information.

Here's a simplification of a basic *Prom Week* instantiation (most of ours were longer than this). Rather than specific characters, all actors in these scenes are labeled Initiator (I), for the person attempting the action, and Responder (R) for the character accepting or rejecting it.

Generic Stop Dating (Accept)

Initiator (I): Hey, {R}, I don't think we should go out any more.

Responder (R): No, probably not.

(result: {I} no longer dating {R})

This generic scene is obviously not particularly satisfactory. A player who sees the same scene multiple times will have a hard time maintaining a suspension of disbelief that the characters are anything other than automatons. In addition, the scene captures none of the nuance of the rest of the social state, including the specifics of these two characters, other than the fact that they are participating in this social move at this moment. A human writer, of course, would create a scene carefully crafted to particular characters and a particular situation, using these details to bring the characters to life and advance the story. One basic step we can take towards this is to provide templates to make the lines seem more specific to the characters, which we called "mix-ins."

Templated Stop Dating (Accept)

I: \$(I's word for hello), {R}. Look, I don't think we should \$(I's word for dating) any more.

R: \$(R's way of agreeing).

(result: {I} no longer dating {R})

Were skater Doug the initiator, he might now perform this line as something like "Yo, Lil. Look, I don't think we should be, like, an item any more."

Another approach we can take is to author multiple scenes for some of the most likely situations we imagine this scene taking place in, and giving the system a way to choose which one is most appropriate. (In *Prom Week*, the scene with the highest number of matching preconditions was usually chosen.) Here's a scene for two characters breaking up because one of them cheated on the other, assuming a social world where monogamy is the only possible romantic option.

```
Stop Dating Scene for a Cheater (Accept)
(precondition: I dating R, R dating Other (O))
I: {R}, you {R's word for insult}! I can't believe you cheated on me
    with {O}!
R: It's not what it looks like, {I}! {O} means NOTHING to me! I love
    *you*!
I: Tell it to the hand. We're SO OVER!
(result: {I} no longer dating {R}, {I} angry at {O})
```

A variety of instantiations with different preconditions goes a long way towards conveying the notion that characters are taking actions for a reason.² It makes the character seem as if they have a memory and internal life. Of course, giving them an actual memory is a logical next step. *Prom Week* stores every social action taken, and the presence or absence of certain types of actions can be used as preconditions on scenes, and referred to by characters in a way that will be linguistically correct regardless of who's speaking to whom. For instance, imagine we'd added the additional effect to the above scene:

²They are, of course: this scene would have happened because of A's volition to stop dating B, which may well have been because of B's unfaithfulness. However, in *Prom Week* we didn't actually tie selection of scenes to matching rules in the volitions: the system's choice of action and choice of narration to justify that action were independent. Surprisingly, this worked just fine in the vast majority of cases, perhaps because real humans are also not the best at giving the most accurate justifications for why they are doing things.

```
(result: remember that {O} did something mean to {I}: "{O} stole  
{I}'s {partner}")
```

Now, a later scene might be selected because it references such an event. Imagine that later, the cheater (now the initiator of a new action) tries to become friends with the heartbroken character from the earlier scene:

```
Bad Blood Start Friendship Scene (Reject)
(precondition: {I} did something mean to {R})
I: Hey, {R}, I was wondering... you want to hang out after school today?
R: Yeah, that's never going to happen.
I: Uh, why not?
R: You think I've forgotten about that time when {the mean thing}? Die
   in a fire.
(effect: {R} is annoyed with {I}, {I} is sad)
```

The responder's devastating line might be rendered "You think I've forgotten about that time when you stole my boyfriend?" Now our system can indirectly narrate cause and effect, surprising reversals, reintroduction of past plot material, grudges and reminiscences and embarrassing moments that can't be lived down, with the help of a unique library of content for each player that expands the more they play. It goes a long way towards gluing a bunch of individually authored moments into something that's starting to approach a story, without creating the combinatorial explosion you'd get from a branching conversation tree.

Another technique we employed was a "cultural knowledge base" of items in the world, and both a personal and universal way characters could relate to them. For instance, a *samurai sword* was an item a character might *have*, *want*, or *dislike*, independent of the fact that characters knew it was generally considered to be *cool*.

Creating an appropriate cultural knowledge base and giving characters connections to items within it gives them things to talk about. For instance, we could write a scene where two people become friends because of a shared interest:

```
Shared Interest Start Friendship Scene (Accept)
(precondition: {I} has {item}; {R} likes {same item}; {R} does not
  have {same item})
I: Hey, I heard you like {items}?
R: Uh, yeah I do! Why?
I: Well, I've got one. If you want to hang out sometime I'll let you
  borrow it.
R: Oh really? Cool!
(effect: {I} and {R} are now friends)
(effect: remember {I} did something cool to {R}: {I} let {R} borrow
  {their} {item}.)
```

This also lets us throw in character-specific details, such as, for instance, someone making a derogatory comment about how “{I} is so obsessed with {item}.”

Put together, all these techniques went a long way towards bridging the gap between reusable lexia and an instantiation that seemed tailor-made for a particular moment.

6.1.2 Templates and History

As seen above, *Prom Week*'s social simulation is given more narrative specificity and context by extensive use of a templating system that can refer to recent events. A historical record saves each action taken along with optional tags about the social significance of the action (such as something being *mean*). Rules can refer to history events in the abstract (*A person is less likely to become friends with someone who has been mean to them*). The templating system can then reference a matching historical

event in dialog. A line using history to justify a rejected Ask Out move might be written as “that time when [mean(responder, initiator)]” and be filled out in play as “that time when you cheated on me.” This use of history to both influence and justify behavior is crucial to helping *Prom Week* characters seem more complex: it gives them the illusion of memories and complex behaviors like holding grudges.

The templated dialog system more generally is critical to *Prom Week*’s success at telling socially emergent stories with natural language. The system’s job is to narrate a change to the social state. *Prom Week* has 39 unique actions: several variants each on twelve types of actionable state changes, such as *begin friendship*. Since each action can be accepted or rejected, 78 scenes are required for the system to be able to narrate every possible action result. However, over eight hundred scenes were authored in practice, with varying preconditions specifying likely social states in which that action might take place. For instance, a more specific *begin friendship* scene might be one in which two characters connect over a shared interest; another might be one where two people become friends despite a negative event in their past.

Of the variant scenes available for a particular social move, the system picks the most specific available to perform,³ then further customizes it to the current situation through templating. In addition to referring to past events, templates handle more prosaic issues such as filling in correct names and pronouns, and can also make use of character-specific slang, or cultural knowledge base connected to an exchange (such as the shared interest that inspired a friendship). Templates were used on average 1.1 times

³Specificity is defined through the number of matching preconditions; scenes are also weighted by other factors, such as how recently this scene was last used.

per line of dialog, meaning there were frequent opportunities for the system to customize a scene to the player’s unique social situation, and details brought in by the templating system in general went a long way towards making scenes seem as if they were authored for specific characters.

While most templating code was fairly straightforward, narrating events from the history correctly given the complexities of language proved more complex, requiring in the end over a thousand lines of code. An event had to be describable by independent parties, both parties involved, one party speaking about themselves or another, including in cases when there were three possible roles; which meant the system needed to understand gender, direct versus indirect objects, rules for English lists (such as the order in “Monica and Zack” not mattering, while it does in “Zack and I”) and many other small details. The result was important, however, in that it allowed characters to reliably narrate history events without sounding artificial: specifically, without players realizing a template was being used. When Edward can naturally rattle off a memory about “that time you and Monica saw my friends and I at the mall” and not realize the complex natural language processing going on to produce this sentence,⁴ it goes a long way to make the templated dialog seem more believable.

A frequent source of bugs was incorrect template text or rendering code, resulting in one of the many kinds of ways English can go grammatically wrong, such as “Recently mine parents went to France...” (caused by not using the right kind of possessive pronoun tag). Related was an issue where templated content only worked if

⁴Note that this phrase might with a different speaker have been rendered “that time Zack and I saw Edward and his friends at the mall.”

used in a particular way that was difficult to remember at authoring time: our cultural knowledge base items, for instance, had to take the form of specific, portable objects that could be pluralized by adding an “s” to the end, or risk showing up in contexts that didn’t make sense. A tool that would show live previews of a piece of templated text rendered in many possible ways would have helped with this.

6.1.3 Other Writing

While scenes made up the bulk of *Prom Week* writing, the design of the game called for a nontrivial amount of additional content to be produced:

- The social schema, including everything simulated by the social state
- The rules by which character actions can change the simulation
- The game’s eighteen characters and their definition, including their traits, their set of starting opinions towards each other, their custom mix-ins, and so on
- A “back story” consisting of a set of history events between characters with negative time steps, so characters could from the beginning reference past events as justification for actions
- The cultural knowledge base and how each character relates to all the items in it
- A set of “campaigns” centered around eight characters, taking the player through a series of levels designed with specific goals for the player to try to achieve; each campaign included introductory text for each level, and a variety of custom ending scenes depending on which goals the player accomplished.
- A set of “text messages” that characters would use to idly communicate between player moves, as a way of performing the current state.

All together, this made up a significant amount of extra content that needed to be created. While the first two items in this list are necessary components of most

social simulations (5.4.1), the rest were specific to *Prom Week*, and each was designed to help players better understand or operate the simulated world.



Figure 6.1: Text messages communicating the social state in *Prom Week*.

The text messages are an illuminating example. Added later in the design process, the idea behind these was to have a running ticker on screen of idling characters reacting to the current social state, as a way of making this more visible to players. So, for instance, one message might be:

```
(precondition: {I} is dating {R})
I: OMG, my {boyfriend/girlfriend} {R} is sooo {I's way of complimenting}!
    Just thought you all should know. <3
```

The system randomly looks for aspects of the current state it can express through one of these lexia, and pushes one to the screen every couple of seconds. We found this was highly effective at making the characters seem more lifelike and aware of the current situation, and also reminding players of aspects of the current state they might have overlooked or forgotten. Because these were single lines without interaction, a change in state, or a realized performance with the game’s animation engine, they were much quicker to author than normal scenes. The final game included over two hundred of these text messages, with the ability to narrate many potential aspects of the simulation.

A more difficult component to author was the starting state and back story. A single character in *Prom Week* could theoretically have nearly five hundred data points defining them,⁵ and while characters never approached this theoretical maximum, upon the game’s release the average character had roughly sixty such data points: feelings towards other characters, personality traits, and connections to cultural knowledge base items. For our eighteen characters, that meant over a thousand individual data points went into defining the initial social state to create a believable social network when the game began. Worse, our decision to keep this state consistent across different scenarios meant that changes to this initial state had to be carefully considered: adding a relationship between two characters, for instance, could break a scenario designed to get the two of them together. In addition, this also does not count the “back story” of history items for each character, so they could reference events with each other in

⁵If they had opinions about and connections to everyone and everything in the game, and every possible trait and status.

conversation: nearly a hundred of these events, meticulously crafted so the template system could narrate them correctly and synced with each other on a timeline, were also created. In hindsight, a system that randomly generated a starting state, or allowed a custom state for each scenario, would have been much less work. (Even more procedural is *Redshirt*'s approach of not having fixed characters at all, but generating them and the starting social state from scratch with each new game.)

6.1.4 Surprises

Many of the design lessons learned during *Prom Week* were laid out in the previous chapter (5.3.1). However, here are a few surprises and their solutions specific to authoring that came up during *Prom Week*'s development.

Rules at multiple granularities. *Prom Week*'s temporary statuses and permanent traits, nearly eighty of them together, laid out a robust set of conditions and personality types that could apply to our characters to help the simulation tell stories about them, including concepts like being *clumsy*, *competitive*, *angry at* someone, a *class clown*, a *bully* or *heartbroken*. However, as we began to fill out the content library we realized this level of specificity was a double-edged sword. Writing a scene for just one of these labels made it a very small slice of the simulation space, meaning it would only appear a correspondingly small percentage of the time. We quickly determined we needed a way of addressing the simulation state at a lower level of detail, to make broad generalizations about categories within the social milieu.

We addressed this by adding the concept of “status categories,” special labels that would match when any of their children were true. For instance, the status category *feeling bad* included the statuses *embarrassed*, *shaken*, *desperate*, *grossed out*, *anxious*, *sad*, *confused*, and *lonely*. This made it much easier to write a single scene that required or referenced someone “feeling bad” and see it show up in the game. We similarly defined ten “trait categories.” The ability to label history events with a subjective label, while designed into the system from an earlier date, also helped serve this purpose: we could write scenes less about specific actions and more for actions that had been labeled *cool* or *lame* (there were thirteen such labels in total). This ability to zoom in and out of specificity was highly useful, and something I would design in from the beginning of a new social simulation engine if I started working on one today.⁶

The system could perform our characters. The positive flip side to the above was that once our library of instantiations was in place, it was thrilling to see the system correctly perform characters we had only defined through aspects of our schema. Our characters exhibited distinct personalities, which was incredible since no specific lines of dialog had been written for any of them. Rather, the system knew that a particular instantiation was *the best way* for shy, dumb Buzz to ask someone out, or vindictive,

⁶In one other interesting permutation on this, we ultimately found we needed additional “invisible” traits that captured qualities only extant in the character artwork. For instance, authors found at one point that they wanted to write a line referencing another character’s funny hat (added by the artist), but realized the system had no way of knowing who was or wasn’t wearing one. We used just three of these invisible traits (*wearing a hat*, *cares about fashion*, and *muscular*) but they went a long way in adding details that seemed more customized to certain characters, and were an interesting example of the art and design iteratively influencing each other during the game’s development. (The characters’ starting traits and statuses were themselves largely based on our subjective reactions to the character art, which was mostly created by artist Kathleen Kralowec without specific guidance.)

shallow Monica to try to be impressive. The schema we'd created and the lexia we'd written to narrate changes to its state provided a canvas on which the system could accurately render our characters, which was very exciting to see.

State changes as storytelling tools. Each *Prom Week* instantiation is conceptually an atomic unit that narrates a single action with one or more concrete changes to the social state. We had always planned to show a “summary” at the end of an instantiation that explained what happened and its results, so the player understood its effect on the simulation. What we hadn't fully considered is that each scene would have its own arc of ups and downs, and that phrasing those in the same ludic language could be used as another storytelling tool.

We added in our authoring tool the ability to tag specific lines in an instantiation with “partial change rules.” These looked like any other kinds of state change (increasing a friendship value; adding or removing a status) but with the key difference that *they were not actually applied to the state*. Instead, partial changes were shown to players during the scene as animated icons above their heads (like the floating *1UP* bonuses in arcade games). We kept as an authoring convention the notion that these partial changes would add up to the final state change by the end of the scene—but their inclusion incrementally along the way opened up a new storytelling tool. Suddenly we could play for humorous effect the notion of a character slowly increasing their affinity for another until the floodgates open at a pivotal moment with a huge affinity boost; or an emotional breakup between flighty characters where they appear to start and stop

dating multiple times within the same scene. The partial changes let us use the language of the simulation as another storytelling tool, and helped a great deal to connect the purely narrative content of the scenes with their ludic consequences.

Overauthored. As has perhaps been implied already, in hindsight we suspected that we ended up creating more content for *Prom Week* than was strictly necessary. In many ways an experimental game, when the team began work on the project there were very few existing examples to compare ourselves to: we had no idea how much content would be necessary to effectively narrative a simulation, or how complex such a simulation needed to be to stay interesting. Many of our the nearly eighty traits and statuses laid out at the project’s beginning were not strongly distinguished within the rules and authored content, for instance, leading to workarounds such as the trait and status categories described above. In hindsight, this was perhaps a sign that we didn’t need as many specific traits as were initially created. Similarly, the final game had on average 3.25 actions per intent and 24 templated scenes per action, which provided a lot of run-time variety, but also much overlap between similar actions: a smaller number of scenes with more complex templating might have provided a similar level of variety with less authoring effort.

The limitations of simulation. A curious form of bug that appeared during *Prom Week*’s writing was one in which a character would say something demonstrably untrue, or that would be contradicted a few scenes later. This was often a result of the simple fact that the lexia were written on top of a simulation that, while complex, still only

considered a small fraction of all the possible things the dialogue might be about. For instance, casually writing a line like “We should spend more time together” could be dangerous: the system might deploy this line for two characters who have been constantly interacting for the last ten moves. An instantiation where someone complains about math might be picked for a character who just claimed they loved it in another: since feelings towards school subjects was not part of our simulation, the system would have no idea the two statements are inconsistent.

In each of these cases we could have added something to the simulation to capture the extra information—extended the schema to add another simulative register—but that would almost certainly have been the wrong decision. A more complex simulation would have needed even more authoring to cover its possibility spaces, which in turn would likely lead to more of the same kind of problems. Instead, a more careful form of writing must be applied: one where lines are more thoughtfully considered. Letting the performance flow from the underlying state we know to be true (from the preconditions) rather than drifting into digressions, or avoiding establishing concrete facts other than those that come from the simulation, are two techniques that can work for this. Likewise, ensuring the scene has all the preconditions it needs is crucial: we once encountered a bug where a character counseled another about how to deal with a cheating partner, neither referencing the fact that the character doing the counseling was in fact the cheating partner. An additional precondition on this scene fixed the issue.⁷

⁷Many *Prom Week* bugs could have been spun off into interesting off-Broadway plays, such as one where the same character was duplicated across all the people in the level, leading him to have introspective conversations with himself about why they weren’t friends.

6.2 Ensemble: A Design Tool

Using the lessons from *Prom Week*, Ben Samuel, Paul Maddaloni and I reimplemented CiF a few years later in a new language and with additional features, in a project that would eventually be named the Ensemble Engine. This work had several major goals: (1) to move development from the dying ActionScript language to a more active and portable JavaScript implementation; (2) to have a fresh code base for an industry-sponsored research project to prototype how social simulation might be integrating into an existing AAA game franchise (described in Appendix A); (3) to generalize and extend CiF to remove specificities that had snuck in during *Prom Week*'s development and enable it to drive future social simulation games; and (4) to move towards releasing a documented and open source social simulation engine.

In addition to helping with the rewrite, one of my major contributions to the project was the creation of an authoring environment for Ensemble, called the Ensemble Tool (Figure 6.3). The tool was deliberately designed to be a second-generation tool for authoring social simulations, taking into account many of the lessons learned creating content for *Prom Week*. I will describe both the engine and the tool below.

The Ensemble Engine is described in detail in (Samuel et al., 2015b). In brief, the project keeps the same core design as CiF, but streamlines, generalizes, and extends several key aspects. Specifically, we (1) made the engine completely domain-agnostic, allowing designers to specify any kind of social schema they would like; (2) allowed rules to be written for any number of character roles instead of a maximum of three; (3)

redesigned actions to be hierarchical and nested, allowing for the design of arbitrarily complex action structures; and (4) included more built-in support for diagnosing and debugging the social simulation’s behavior. The code was rewritten from scratch in Javascript (in the hopes it would be useful to the fast-moving, innovating gamemakers behind, for example, the Twine revolution (Ellison, 2013)) and is planned to be released as an open source library.

The flexible schemata allow Ensemble to drive social simulations at a range of fidelities. A schema specifies one or more categories, which represent a particular way of representing social information along with a series of types instantiating it. For example, *Prom Week*’s built-in notions of characters having variable numeric feelings towards each other could be recreated in Ensemble with a category called “feelings” with the qualities directed, numeric, and no duration, and with the three types *friendship*, *coolness*, and *attraction*. Ensemble’s system also allows creating new ways of representing a social or game state beyond those

seen in CiF. For instance, a traditional roleplaying stat could be created by making an undirected numeric stat that ranges between 3 and 18.

The Ensemble Editor also represents a significant improvement over the editor

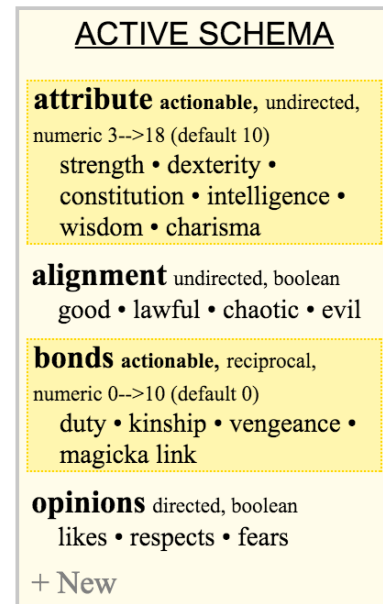


Figure 6.2: A sample schema for a role-playing domain in Ensemble, showing custom types defined by parameters (such as *actionable*, *undirected*, and *numeric* for **attribute**).

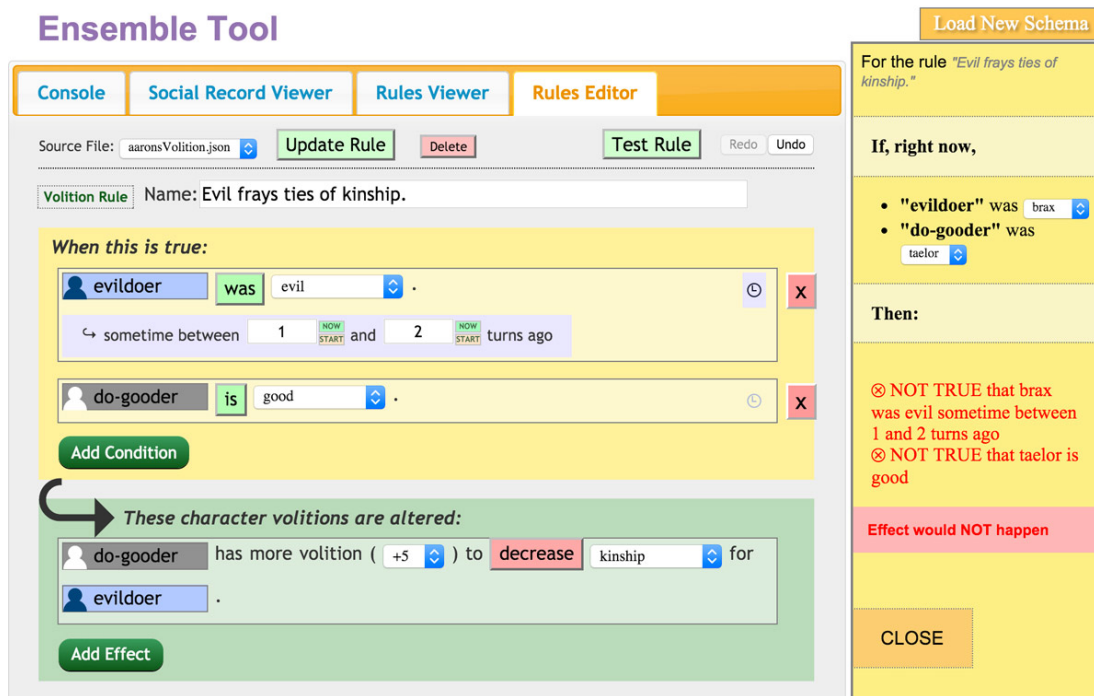


Figure 6.3: The Ensemble Tool: Rules Editor view.

used during the authoring of *Prom Week*. The tool allows for creating and modifying a schema, trigger and volition rules; testing rules; viewing a history; and a console that allows for experimentation with the social state. We had several key design goals during construction of the tool.

Make rule authoring more visually understandable. An Ensemble rule is a complex set of predicates: one or more conditions and one or more effects, any of which may have temporal restrictions and each of which will have different data depending on the referenced schema category. The Ensemble tool displays rules in a visually clear format (Figure 6.3) that alters itself based on the active category, hiding fields that aren't relevant to the specific predicate. Instead of fixed placeholder labels, roles within

a rule can be given variable names to help track logic across a rule: each name receives a distinct color in the editor. Rule predicates are automatically summarized in natural language in a list viewer, which allows for quickly seeing a dozen or so rules at a time; this view can also be filtered by any term appearing in the rule. These features reduce the cognitive load on authors and hopefully will make rule authoring a more pleasant, tractable experience.

Allow for testing within the tool. The Ensemble engine runs within the tool, allowing the author to test rules live as they code. A console mode allows changing the state of the embedded simulation and manually triggering behaviors such as advancing the time step or recalculating volitions. A pop-up in the rules editor allows for binding roles to particular characters within the schema definition to see if a rule would currently hold true given the simulation's current state. Enhanced error checking also significantly reduces the likelihood of malformed rules that might cause a run-time crash.

Minimize the need to manually update the tool. The new tool is designed to whenever possible build UI elements around the schema and rules the designer has defined, rather than maintain a separate definition file that risks falling out of date. Drop-downs, generated rule descriptions, and rule editor UI components are all generated live from the current schema, and immediately update if that schema is modified. These features make it less likely the tool will fall out of date as the social simulation parameters evolve during normal game design iteration.

The Ensemble tool represents a significant step forward in freely accessible

tools for social simulation, and along with the Ensemble Engine make up two major pillars that a designer would need to create a new social simulation storygame. We hope to provide another of the remaining absent pillars (integration with one or more existing game engines) in the near future.

The other major work I have done with social simulations is the previously mentioned Project Yarn, a pilot project with an industry team to explore what implementing social simulation into more traditional game mechanics might be like. This project is described in detail in Appendix A.

Chapter 7

Collaborative Storygames

Is Dungeon Mastering an art or a science? An interesting question!

Foreword, Dungeon Master's Guide, 1st edition

If there were a theory of the fantastic like there is in the case of logic, then we would be able to discover the art of invention.

from Fragments by Novalis (1772-1801)

The final storygame genre I wish to discuss is *collaborative storygames*, for which the lexia are generated at least in part at runtime by the participants, assisted by systems designed to produce a compelling story. While this genre includes familiar non-digital games, such as the tabletop roleplaying game *Dungeons & Dragons*, it can encompass a broader range of analog, digital, and hybrid works than are generally discussed together.

Consider, for instance, the 1993 game *Tales of the Crystals*, released by Milton Bradley and presumably targeted (see Figure 7.1) at preteen to early teen girls. A group of two to four players gather and listen to an audio cassette tape narrating the adventures of a group of magical creatures in a fantasy realm. But on occasion these

characters need help from a group of heroes, the Keepers of the Crystals: the players. The cassette tape narrates a problem that needs resolving, then asks the players to stop the tape and devise a solution. Unlike other contemporary games with audio or video tapes, the solution is not devised through a set of board game mechanics, but in freeform discussion between the players: they make it up. When the tape starts again, it cannot (of course) respond to what the players have decided, but nor does it contradict their solution, continuing on as if the problem has been resolved but leaving out the particulars—the pieces of story which the players have just invented amongst themselves.



Figure 7.1: Back of the box for *Tales of the Crystals*.

The game provides a number of tools to stimulate the players' imagination:

to help them create more satisfying stories about the Keepers. The narration on the cassette is filled with evocative descriptions often directly addressed at the listeners, asking them to picture things or imploring them to take physical actions: “Place your hands upon your crystal pendants, and see if you too can feel the danger,” a character says in “Save Collingwood,” the tape’s first adventure. The pendants are real props supplied with the game, along with an array of other material: cards providing ideas or extra challenges, with backward text that an included “magic mirror” can decode; a journal to take notes about problems and the agreed-upon solutions. The taped narration hints at various complications and possible solutions to the problems the Keepers face, providing raw material for their discussion. Finally, the division of players into four possible roles (healer, protector, scribe, and leader) allows characters on the cassette to address specific players, gives each player the chance to have a unique role in helping solve problems, and provides a way of mutually reinforcing the shared fictional reality that might be much more difficult to sustain in a game made for only one player.

Crystals is a useful entry point into our discussion of collaborative storygames for several reasons. It reminds us that the audience for these kinds of games (potential, current, and historical) is broader than the geek stereotype they’re often associated with. It also reminds us that while these games are often romanticized as “pen and paper” alternatives to technology-driven interactive stories, tech has always been intertwined with these “analog” games in deep and rarely considered ways, from the need for accurate random number generation that spurred the regular manufacturing of numbered polyhedral dice (Peterson, 2012, p. 304) through to the modern proliferation of tools

for generating fantasy names¹ or allowing remotely distributed players to move figures on a virtual battle mat.² Finally, through it we can see that tropes we might assume are standard in the genre (such as the need for a gamemaster, or rules for numerically simulating a fictional world) are not in fact foundational, setting us up for the same bottom-up construction of a conceptual framework that we have done for other kinds of games in previous chapters. We are working not to define the components of *Dungeons & Dragons* and similar games, but to find the conceptual loam in which games of this sort can grow—and uncover what other kinds of games might grow there as well: in particular, what kinds of digital games might sprout from similar seeds.

I'll start by defining what I mean by collaborative storygames, and how my use of this term overlaps with existing work in the study of tabletop roleplaying, author-assist tools, and systems for computational creativity. I will then describe a design framework that places these games at the middle of two spectra (a simulative and a performative), and defines when a gamemaster is necessary to mediate between each spectra's extremes. I will then consider several analog games that show the interesting design experiments happening in this space. Several digital games will then be considered as forays into how these insights can be applied to computational systems. This will in turn beg several questions that need discussing in order to round out our theory, including *What does it mean to collaborate with a non-sentient system?* Finally, I will discuss several promising futures for digital collaborative storygames, including a design methodology by which both the shared language of sculptural fiction and the open-ended possibility space of

¹Several hundred of which are catalogued at <http://www.fantasynamengenerators.com/>

²Such as in the system Roll20 (<http://roll20.net/>)

social simulation can help point towards new ways of interacting with a digital story. As before, the following chapter will include a case study of my own experiments in both analog and digital collaborative storygames.

In this work my aim is not to discover a replacement for human gamemasters to enable better computer RPGs, nor to “disrupt” the space of tabletop roleplaying. Rather, I aim to suggest that there is a third kind of game that can arise from studying both digital and analog storygames together and deeply understanding the conceptual framework that underlies them both. Teasing out the parameters of this nascent design space will be the primary design contribution of this chapter, and describing how emerging technology can enable new spaces of design within it will be its technical contribution.

7.1 Definition

As in previous chapters, let’s begin by establishing a definition for collaborative storygames, specifically making sure we include both traditional tabletop roleplaying but also other kinds of storytelling systems. It’s useful to once again contextualize this within our broader definition for a storygame, which we recall is:

- a playable system, with
- units of narrative, where
- the understanding of both, and the relationship between them, is required for a satisfying traversal.

A collaborative storygame, then, has two additional constraints:

- some or all the lexia are created during play by participants, and
- that creation is mediated by systems designed to produce a compelling story.

Created during play. In traditional stories, lexia are created “offline” (before narration time) by the storyteller and told in a fixed, linear order. In digital interactive media, the lexia are generally also created in advance, though the order in which they’re revealed to a player can depend on their decisions and the narrative logics of the system. Some frameworks we’ve considered, such as social simulation, can lightly or even extensively customize lexia for a particular situation. But collaborative storygames are the first type of storygame we’ve considered where lexia can be wholly created during play.

Humans, of course, are good at spontaneously generating stories (even untrained humans who might deny they have this skill; see for instance Samuel 2016, p. 136). Most computer storytellers try to simulate a responsive human storyteller by having an extensive library of pre-authored lexia created by a designer, in the hopes of having responses to as many of the player’s potential inputs as possible.³ In this chapter, we would like to make a distinction between this form of pseudo-collaboration, and genuine responsiveness: actually creating a lexia in response to input from one’s collaborator.⁴ I

³Recall one of the failings of the eureka story in adventure games (2.2), when the game system fails by not having a lexia matching the player’s attempted contribution.

⁴Consider, for example a gamebook like a *Choose Your Own Adventure*. We might reframe a player’s interaction with it as a collaborative storygame, with the pre-written system and the player’s spontaneous reaction two halves of a collaboration:

Book: Perhaps you’d be interested in a story where you choose whether to go down the left or right tunnel.

Reader: Hmm, perhaps in the story I go down the right tunnel.

Book: Ah, then perhaps you should turn to page 54 to see where it leads.

However, this frame seems an awkward fit, in part because neither the player nor the system can provide an expressive response to the events of the story. The system is constrained by what’s been

will frame this in (7.5.2) as the difference between naïve and deep collaboration.

Participants, plural. An implied part of our definition is that collaborative storygames need multiple participants. Collaboration means a give and take of ideas from multiple sources to produce a single unified output: this cannot happen alone. This does not necessarily mean we wish to restrict our definition to multi-player games, however, because I would like in this chapter to consider the possibility that one or more participants in a collaborative storygame might be a system: a set of rules or a computer program.⁵ However, most existing collaborative storygames are designed for multiple humans to play, and this is the baseline from which the perhaps radical assumption above will spring.

written, and the player by the limited set of available choices. This deception is made more obvious if we remove the choice:

Book: Perhaps you'd be interested in a story where Vorgo continues down the tunnel.

Reader: Yes, I think I'd like that.

Book: Ah, then perhaps you should turn the page to see where it leads.

This is clearly less of a collaboration and more of a shell game. But consider a similar sequence in a tabletop roleplaying game:

GM: So Vorgo continues down the tunnel...

Vorgo's Player: Wait. Does it look like anyone's been down this tunnel recently?

GM: Hmm... yes, actually, you find a small ruby scale on the ground.

Player: Salamander folk! I start moving more quietly...

If we imagine the gamemaster has invented this detail of the ruby scale, we see the key difference: they created something genuinely new in response to an unexpected player action, and the player in turn reacts to that with another novel behavior. This is collaboration in the sense we mean, while the gamebook examples are not.

⁵Interactive story researchers might immediately assume we are talking about the field of story generation research (briefly summarized in 5.2) but I mean to include a much wider variety of things in this definition. Specifically, later in the chapter (7.5.2) I will break down the difference between generating ideas and creatively combining them, and posit that a system that does the former paired with a human that does the latter can be a form of collaboration.

Systems to produce a compelling story. The second criteria for collaborative storygames is that the act of generating and synthesizing lexia must be mediated so as to produce a compelling story. This distinguishes a number of games and playful activities that involve creating story elements from the kinds of games we wish to discuss here.

For example, consider the storytelling card game *Once Upon a Time*. In this game, players are dealt a hand of common fairy tale tropes, and must attempt to tell a story that uses each of those elements, playing the card to the table when they do so. The person who can play their final card while narrating an ending to the story wins. However, other players can steal control of the story if someone mentions a card that they're holding. While the rules say storytellers should avoid things that are "foolish" or "contradictory," there are no mechanics that enforce the notion of a coherent or compelling story, and indeed the mechanic of pitting players against each other discourages collaboration or working to tie story threads together. The system does not help players tell compelling stories, which places the game outside the space we're primarily concerned with here.

Another flavor of games outside this space is those for which a compelling story is not the desired goal. Many improv theater games, for instance, have goals such as making the audience laugh or helping actors think on their feet, and might include mechanics specifically designed to make stories *less* coherent but more entertaining by playing the former goal off the latter (such as asking the actors to continually incorporate random suggestions from the audience into an ongoing scene; see Levy 2005).⁶ A

⁶There are theater games that do focus more on coherence: see, for instance, Ben Samuel's fascinating description of "The Form" (2016, p. 147).



Figure 7.2: A selection of cards from *Once Upon A Time*.

distinguisher between these two types of games is the presence or absence of an audience to be entertained: whether the primary pleasure is meant to come from consuming or creating the story.⁷ When the audience is passively watching, improv’s “yes, and” rule (which states that performers should generally incorporate whatever their scene partners say into a scene without contradicting it) works well in a context where stopping the action to discuss the merits of a contribution would be poor form. In tabletop roleplaying games, however, this discussion over potential futures before choosing one to enact is

⁷Roleplaying game designer Robin Laws notes that:

... the gaming experience itself is not set up to be observed by outsiders. Unlike the other traditional narrative forms we have been drawing analogies to, gaming does not draw a line between artist and audience. In a gaming session, all participants are creators. They are not passively watching a predetermined work of art unfold before them. They are collaborating together to create a work that exists only for a moment, without the eyes of non-participants upon them. (1995)

often the meat and bones of play (Drachen and Smith, 2008).

7.1.1 Connection to tabletop roleplaying theory

Tabletop roleplaying games are in a sense the closest medium so far to the dream of radically interactive storytelling— a paper prototype or “Wizard of Oz” approach to experimenting with stories that respond dynamically to player input. Given this, it’s surprising how little research has been done by interactive story researchers into their form and structure.⁸ Barely twenty years ago, designer Robin Laws opened an influential essay suggesting the then-novel idea that tabletop roleplaying games might be critiqued like other media were (Laws, 1995); not even ten years ago, an academician wrote of the lack of a basic design vocabulary and terminology for tabletop roleplaying games and the “problematic” status of roleplaying theory, saying it “has reached a basic level of academic acceptance, but exists in a state of chaos” (Harviainen, 2009). Multiple theories and frameworks for tabletop roleplaying have been proposed, largely by practitioners (a good overview can be found in Boss, 2008) but these have rarely overlapped with digital games theory or theory from other forms of media, and are thus less useful for our project here.

One exception is the intriguing framework proposed by Lars Konzack, which he calls the Wunderkammer-Gesamtkunstwerk Model for Role-Playing Game Analysis and Design (2015). The second part of this name, “Gesamtkunstwerk” (total work of art), was used by Wagner to describe the way opera brought all the great forms of art of the

⁸Most of the influential design thinking in this space has come from practitioners, not theorists in the academy, perhaps less surprising when we consider the genre’s need for multiple players with large blocks of overlapping free time.

time together into a single, unifying whole. Konzack repurposes this term to mean that tabletop roleplaying consists of a number of unique practices, with antecedents ranging from theater to fiction writing to make believe to military simulation, and a theory of the form should encompass all these kinds of practice holistically, rather than studying them in isolation (citing, for instance, Nordic LARP theory which is based on interpreting live-action roleplaying as a direct continuation of theatrical traditions). Konzack proposes four main components to roleplaying (Sub-Creation, Ludus, Performance, and Narrative), but his larger point is that these various aspects of the form should be studied and understood together.

The other part of Konzack's framework, "Wunderkammer" (wonder-cabinet, or cabinet of curiosities), invokes the Renaissance tradition of assembling collections of unique objects for display. The Wunderkammer creates a magical space where objects take on special significance by virtue of their inclusion, and the viewer is invited to make connections between them: "a microcosm of wonders that trigger imagination and ingenious thoughts as to the greater macrocosmos of which it is an representation." The collection of objects is presumed to stand in for a larger whole, extrapolated from the cabinet's contents. Konzack draws a parallel between the magical space of the wonder-cabinet and the "idea space" that tabletop roleplayers create together in play. As objects or ideas are added to the world through verbal description, players are continually revising their concepts of the shared fictional universe, based on their interpretation of how everything in the Wunderkammer of ideas stands in for a shared fictional world.⁹

⁹Konzack gives the example of introducing a crystal skull into a game of *Dungeons & Dragons*, with its stock fantasy setting, versus into a game of *Call of Cthulhu*, based on the supernatural horror of H.P.

This process of synthesis, which has also been called “resolution” (Boss, 2008) is one of several key activities that collaborative storygamers can engage in. In the framework developed below, I label this Storywrighting. It is the activity computers are least well suited to engage in, a point I will return to throughout the remainder of the chapter as we look for ways that system might contribute to collaborative storygames.



Figure 7.3: A modern Wunderkammer.

7.1.2 Connection to existing collaborative fictions design work

Collaboration has been discussed in a slightly broader context in the fields of computational creativity, and in guises such as mixed-initiative design tools (Smith et al., 2010) and shared authorship (Samuel, 2016). The latter comes closest to our notion of collaborative storygames. Samuel writes that a work of shared authorship “is one in which the player and the system collaboratively create a narrative artifact, ideally one which neither would have been capable of producing on their own.” He proposes a

Lovecraft. Players are likely to interpret the object’s particulars and significance in very different ways (perhaps as valuable treasure in the former setting, and a dangerous supernatural artifact in the latter), based on their understanding of the world they’ve created in their play sessions. That understanding is in part created by past objects which have been brought into their conceptual Wunderkammer. But the introduction of this object, in turn, will modify that understanding (i.e., “this setting is like *Cthulhu* but with crystal skulls”).

tool that uses several computational techniques (including social simulation and play trace analysis) to allow a human author to play with possible stories and receive sensible suggestions for things that might happen next. Samuel's project builds on a small handful of predecessors such as *Say Anything* (Swanson and Gordon, 2008, 2012) which attempt to provide feedback to natural language human input in the hopes of sparking human creativity. These projects share similar concerns to those under discussion here, but differ primarily in that they are tools for creation, not games of creation. Recall that a storygame involves the satisfying traversal of a narrative, even if that narrative is being spontaneously generated on the fly; tools instead involve iterative, analytical improvement: a detached perspective rather than an embedded one.

However, the distinction is subtle enough that many lessons from these projects can be reapplied here. Samuel, for instance, catalogs the pleasures of shared authorship, many of which also apply to collaborative storygames, such as the feeling of trust that develops when one feels understood by one's collaborator, and the feeling of coming to understand and learn about that collaborator in turn (2016, p. 28). Later in this chapter (7.5.1.1) I'll discuss Emily Short's *Annals of the Parrigues*, in which the author and her digital collaborator's understanding of and trust in each other was a key part of the creative process.

7.2 Design Framework

While these prior approaches inform our thinking, collaborative storygames are a somewhat different animal, and deserve their own design framework.

I begin by placing collaborative storygames at the center of two spectra of behavior (Figure 7.4): a simulative spectrum and a performative spectrum. I will discuss how the role of gamemaster traditionally serves as a mediator between the edges of the spectra, and is necessary when there is an imbalance of knowledge about the rules, the story, or both. I then suggest that participants of collaborative storygames are engaging in one or more of four major categories of action: Generating, Storywrighting, Administration, and Negotiating.

7.2.1 Simulative Spectrum: World to Story

Let us consider an axis for games that tells us whether their rules are primarily concerned with logic and simulation or telling a story. On the logical end we might place a game like chess, with a veneer of narrative but no rules to help create it. On the other end, we might place immersive live-action roleplaying (LARP) with no gamemaster or mimesis-breaking rule mechanics, with a veneer of simulation (a prop standing in for a more grandiose version of itself, for instance) but no simulationist rules.

In between we have many games that are storygames or storygame-adjacent. Moving roughly from the world to story ends, we might place on the spectrum wargaming with miniatures, digital roleplaying games, dungeon-crawl tabletop games, and story-

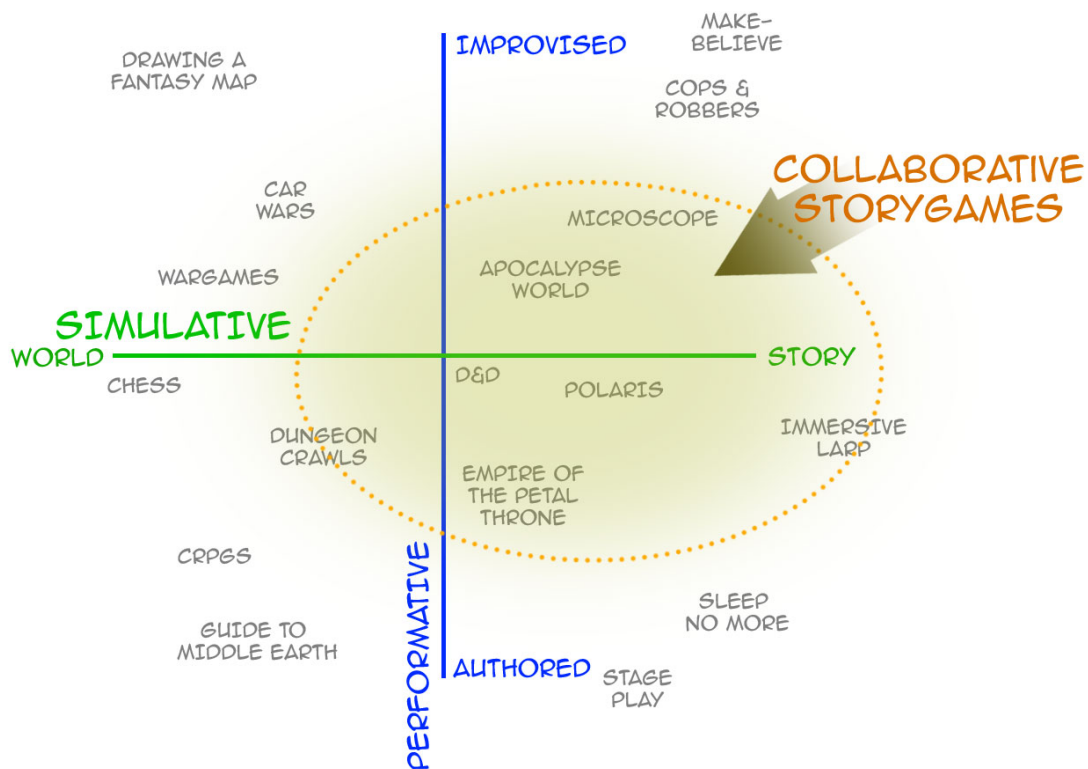


Figure 7.4: Collaborative storygames are at the center of a Simulative Spectrum between simulating a world and telling a story (trending towards the story side), and a performative spectrum between enacting an authored story versus improvising one.

and character-heavy tabletop games. As we move from one end to the other we see an increasing emphasis on performance of character and primacy of story, and a shift from largely ludic rules to largely narrative rules. Another way of saying this is that the games' structures move from simulating a world to simulating a story. Wargames, for instance, may have many rules governing movement, attack strength, and defensive capabilities, but few or none concerning morality, character motivation, dramatic tension, or negotiation. An indie tabletop roleplaying game like *Polaris* (discussed in 7.3.3), however, might have these concerns reversed.

In the center we have games like *Dungeons & Dragons* that attempt to address both ends of the spectrum: simulationist rules for things like movement and fighting, and storytelling rules for things like character alignment, progression of power, and creating memorable villains. Perhaps it follows logically to note that these traditional roleplaying games tend to have much fatter rulebooks than either primarily simulationist games or primarily storytelling games: they attempt to address both ends of the spectrum at once. This productive middle ground has propelled tabletop roleplaying as a major hobby industry for close to half a century, and inspired many of the digital games discussed elsewhere in this dissertation. Clearly, the merging of these two extremes is compelling.

7.2.2 Performative Spectrum: Author to Improviser

Another spectrum runs from completely scripted to completely improvised stories. At one end we have a group of actors performing a stage play from a script: at the other, a group of kids playing make-believe. Again, we have a spectrum of activities between these two extremes, where the amount of agency given to the performer to vary the story itself (beyond the particulars of their performance of that story) ranges from none to all. On one end of the spectrum, all lexia are pre-generated and pre-ordered; on the other, all lexia are generated spontaneously during play.

Again, collaborative storygames tend to be near the middle, combining some pre-generated lexia with some ability to make new contributions and improvise. A game like *Microscope* (discussed in 7.3.2) is nearly entirely towards the improvisational side, while a lore-focused game like *Empire of the Petal Throne* assumes most stories told with

it will incorporate large amounts of existing narrative content from its source material.

7.2.3 Game Master as mediator between two

Simulationist games like wargames may need a referee to explain or adjudicate the rules. Similarly, narrative-heavy games often need someone in the role of a critic, rejecting inappropriate ideas or making a final call as to what ideas become part of the story. *D&D*'s role of the dungeon master¹⁰ was among the first to mix both kinds of adjudication: someone to manage both the rules and the story.

Centralizing adjudication is necessary when there is an imbalance of knowledge or power. For instance, while all players often equally share adjudication of rules in simpler board games, in a game where the rules are complex to interpret an adjudicator may be necessary to ensure all player moves are valid and make unbiased rulings in cases where the rules are unclear or interpretable. Likewise, in a classic roleplaying game, the gamemaster prepares a scenario ahead of time for the other players to encounter, and some of their pleasure comes from not knowing the details of the plot and setting in advance. But in either case, knowledge or power can be distributed across players in different ways, leading to situations where players can share gamemastering duties: many of the games discussed later in this chapter experiment with this in various ways.

Another challenge of mixing these two adjudication roles (referee and critic) is the need to mediate between them, deciding from moment to moment which should take primacy. For instance, a gamemaster might “fudge” the result of a die roll at a

¹⁰I will generally use the more system-agnostic term *gamemaster* or GM throughout.

critical moment because they believe a different fate would make for a better story.¹¹ Alternatively, a gamemaster might decide some aspect of a player's narration should be handled by game rules, either looking them up in the rulebook or inventing them on the spot. Different gamemasters might play at different places on the simulative spectrum: a “dungeon crawl” where very little narrative unfolds and most the players' time is spent in rules-driven combat, compared to a session where the players do nothing but talk to each other in character. This balancing might be set by the game rules themselves, by the gamemaster, or by individual players, depending on how a game is structured: but it exists in some form in all collaborative storygames.

Another form of balance is necessary to mediate the performative spectrum. Certain elements of the game's setting, its rules, and the gamemaster's preparation may be considered static (or templated) lexia that exist when the session begins. How many contributions created during play are allowed, and when—how close the players can get to the improvisation side of the performative spectrum—is another form of adjudication. Too much structure and the players may feel railroaded into an existing plot; too little and the story may lose coherence, interest, and forward momentum. Many games do not provide players or gamemasters with much guidance on how to manage this balance, with the result that many gamemasters are not particularly good at it. Again, some of the games we'll look at later in this chapter (such as *Apocalypse World* in 7.3.4) provide explicit rules for this kind of adjudication.

Computers have traditionally been better referees than critics, meaning the

¹¹The “GM screen” traditionally deployed between gamemaster and players is in part a contrivance to allow this kind of “cheating.”

aspects of tabletop roleplaying computers have encroached on are more biased towards adjudication of rules rather than finessing a story. The rise of story-centric analog roleplaying games over the past twenty years (Appelcline, 2014, p. 137-8) may in fact be a reaction to this. While our project in this chapter is not to further encroach on human gamemasters, we use this framework later on to consider other ways systems might collaborate with players than simply as rules administrators.

7.2.4 Four Roles

While collaborative storygames contain and touch on many kinds of activity, we can consider four major activities that participants in such games are largely engaged in: Generation, Storywrighting, Negotiation, and Administration. These processes take place to a greater or lesser extent in all collaborative storygames. Certain games might provide explicit mechanics for some, or allow the behavior to arise as a narrative dynamic from the existing rules. I will define what I mean by each below.

7.2.4.1 Generation

Generating ideas for stories or worlds is a key part of collaborative storygames: everything from coming up with evocative character names through drawing maps of cities and dreaming up epic plots. These ideas might be created by players during play, or pre-written in the form of back story, character classes, or a set of predefined monsters that come in a game's rulebook.

While this may seem a very basic form of collaboration, Peterson in his history

of *Dungeons & Dragons* talks about the importance of the creators' work distilling ephemeral mythology into concrete statistics:

Where genre authors, who inherited these building blocks of fantasy from myths, could handle these fantastic elements without resolving the vagueness of the legends, *Dungeons & Dragons* forced monsters, spells and magic items to conform to its system, and thus made them specific enough that they could be simulated in a game. The genius of the creative apparatus of *Dungeons & Dragons* is how it lowers the bar for contribution to the fantasy genre: it creates, in effect a do-it-yourself kit, a checklist that prospective monster-makers or spell-weavers need merely fill in with their own fancies. (p. 201)

I'll return to this form of collaboration in more depth under the name "Naïve Collaboration" in 7.5.2.

7.2.4.2 Storywrighting

Mere ideas in isolation are not enough: they must be combined and shaped into an ongoing, coherent, and compelling story, a process I here call storywrighting. This is not just "writing," which does not guarantee coherence or connectedness, but craft: combining ideas from all participants in elegant ways to tell a story. This is the act of curating the Wunderkammer—choosing what to place in it and how what's been put there is meaningful—and it's something humans are especially good at. Recall our discussion in (3.4.2.2) of "fires in the desert," or the story of the doctor driving the Volkswagen bug in Tokyo. We yearn to fill in these gaps between narrative fragments, to make them into a coherent story even (or especially) when they seem contradictory. Gianni Rodari in his book *The Grammar of Fantasy* says that "a story can be born only out of a 'fantastic binomial' ": "The imagination is compelled to set itself in motion to

establish a relationship between the two and to construct a (fantastic) whole in which the two foreign elements can live together” (p. 12). Storywrighting is this process of elegant construction.¹²

When I speak in our collaborative storygame definition of “creating lexia during play,” I am really speaking about both these first two activities: generation of interesting ideas and storywrighting to bring them together. Computers are better at the first than the second; but they can still effectively participate as collaborators, as I will discuss below.

7.2.4.3 Negotiation

When two players have contradictory ideas or want to storywright them together in different ways, there must be a process of resolution by which all participants can agree on a single imagined story space within which play can continue. The game’s rules might provide a form of one-sided negotiation: in early versions of *D&D*, your idea to play a dwarvish magic-user is not allowed because of race and class restrictions. Players often negotiate amongst each other, both in and out of character, about what to do next or who gets what treasure. Players might negotiate with a gamemaster, even if this negotiation is fairly one-sided, or even negotiate over things that have already happened (“Wait, I thought *I* was in front.”) In all cases, the act of Negotiation is the process of determining which possibilities actually happen in the fiction. *Polaris*, discussed in

¹²We earlier discussed *Once Upon a Time*, which is based almost entirely around storywrighting. With this new framework, we might observe that while the system Generates ideas (in the form of its cards), the game has no rules or mechanics concerned with Storywrighting, even though this is the primary activity that players engage in.

7.3.3, turns negotiation over story into a central mechanic.

Storytellers Norma J. Livo and Sandra A. Rietz describe the tradition of oral storytelling as both a game and a negotiation, bound by rules but also contingent on both the storyteller and the listeners to collaborate:

[A]udience and teller negotiate a story into being in a highly dynamic interactive process. The teller may know specific story content that is not known to the audience, but all participants in the telling have a role to play in bringing a story into reality. . . . The role of the teller, then, is to give over ownership of the story to the audience and to allow the audience to encode the story using the teller as the medium. (1986, p. 9)

Storygames, as modern descendants of both the oral storytelling tradition and of games, exaggerate both these qualities inherent in stories that predate the calcifying effects of written words and recorded media. The negotiation in tabletop storytelling is in part a return to a very old and more fluid form.

7.2.4.4 Administration

The last category of actions encompasses all of the scaffolding, logistics, and organization necessary to provide a framework for the other actions to take place within. Most critically, Administration provides rules that structure when and how the other activities can happen. It may also provide procedures for taking part in or responding to them, which might in turn involve rolling dice, writing on or reading from character sheets, creating or looking things up in tables, or resolving rules conflicts. *Apocalypse World*, discussed in 7.3.4, contributes a great deal of Administration designed to help gamemasters Storywright player contributions into a satisfying whole.

7.2.4.5 (Performance?)

Some definitions of roleplaying identify a player's performance of their character as an additional key activity,¹³ but I include this as a form of Generation: contributing ideas about how a specific character (your own) is contributing to the fictional world. Processes of storywrighting and negotiation might also play out as performances in practice: but again, this is just the difference between having these discussions out of character or in character, and the fundamental activity remains the same.

We'll come back and discuss these categories in more depth at the end of the chapter. First, however, I'd like to ground out this discussion by taking a close look at several collaborative storytelling games, considering what this framework can tell us about them and what avenues it suggests for further exploration.

7.3 Example Analog Games

7.3.1 Andrew Plotkin's Journal Writing Game

It might first be useful to demonstrate something that is neither a tabletop roleplaying game nor especially complex: a minimalist collaborative storytelling game.

In 2007, interactive fiction designer Andrew Plotkin created an untitled game¹⁴ that

¹³Peterson identifies two voices that roleplayers often switch between: the immersed voice, where they are speaking in first person and perhaps gesturing or moving as their character, and the detached voice, where they are stating in third person what their character is doing, thinking, or feeling (402). Both are a form of character performance, but with different levels of embodiment. Similarly players tend to shift freely between speaking about events within the fictional world and making extradiegetic comments on what's happening, what should happen, or what's happening in the real world outside the play space: this to me is more a shift of focus rather than a change of activity, from a design perspective (Drachen and Smith, 2008).

¹⁴The game rules are available at <http://www.eblong.com/zarf/uru/rj/essay-more-forum-games.html>.

shows how a few simple mechanics can meet both requirements of our definition: lexia generated at runtime, and rules for producing compelling stories.

Originally designed for two players posting on a public forum, Plotkin's game has each player writing in the same fictional journal from two different time periods: one from a long-dead civilization at the height of its power, and the other as an explorer finding the journal in its mysterious ruins. Each player takes a turn by free-writing a journal entry, and can build off the other's descriptions of discoveries and details in the other time period.

However, a simple currency gives the experience ludic structure. Each player starts with ten white stones and one black stone. On a player's turn, they spend one to three white stones to highlight elements from their most recent entry that they'd like to become important to the story. Players can resolve two or three elements together (explaining how they are connected and narratively significant) to gain black stones. Black stones may be spent to re-introduce a resolved element (saying there's more to the story there) or to contradict something introduced by the other player, who must then explain their mistake in a narratively sensible way on their next turn. Play ends when all elements have been resolved and all white stones have been spent.

These relatively simple mechanics provide the structure and narrative flow to turn a writing exercise into a storygame. The limited number of stones gives the story an approximate length and a way of estimating position within it, and distributes creative power between the players evenly. Players can in theory introduce and resolve their own elements, but putting all introduced elements in the same pool and giving a mechanical

reward for combining them encourages players to riff off each others' ideas, using elements introduced by their partner in surprising ways—recontextualizing them by putting some “next to” others, in the metaphor of the Wunderkammer. In addition to this “yes, and” technique, players can also negotiate (by reintroducing resolved elements) or even veto (by contradicting) the contributions of their partner. But the limited currency of black stones makes this kind of collaboration the exception, not the rule: and the only way to keep doing it is by reverting to “yes, and” collaboration to gain more black stones. Spending a stone to highlight an element is also a form of collaboration, communicating to your partner which elements of the story you're most interested in exploring further.

In this way Plotkin's rules tell us not only what kind of collaboration the game expects, but also how to achieve it. The affordances of the game rules are all acts of collaboration: highlighting a plot point to pursue, negotiating for changes to the story, successively resolving what's been introduced to end up with a tidy conclusion. Plotkin's game demonstrates that a system for producing compelling stories need not be conceptually complex nor contain the human-level intelligence of a professional storyteller, and that not all collaborative storygames descend from tabletop roleplaying traditions.

The next game we'll consider places similar ideas about egalitarian collaboration in a more elaborate framework, radically extending the power of players to build a shared fictional world.

7.3.2 *Microscope*

Microscope (Robbins, 2011) is an unusual sort of roleplaying game. It has no gamemaster, players do not have player characters, and there is no simulated world where time moves forward with each character action. Instead, players take turns building a fictional history, represented as a timeline of index cards on a table. At the start of play, players choose a concept for their history and pick its first and last events. Players then take turns adding new events to the history at any point in the timeline. Events can be long periods of time such as historical epochs, major events within a period, or individual scenes during an event, which are roleplayed out to find the answer to a question framed by the current player.¹⁵ There is no defined end state: play continues until the players are satisfied with their timeline or are ready to stop playing.

The rulebook describes the “vast creative authority” given to its players, one of the game’s most fascinating innovations. While roleplaying without a gamemaster is not an idea unique to *Microscope*,¹⁶ it takes shared authorship among players to an illustrative extreme. Rather than assigning specific functions normally performed by a gamemaster amongst players, here all players are equally responsible for all creative

¹⁵ *Microscope*’s title comes from this freedom to zoom in and out at will, from thousand-year epochs to individual moments; as well as the system’s capability to tell a story spanning billions of years or a single afternoon.

¹⁶ Among the first published GM-less systems was *Universalis* in 2002, where players have a currency that can be spent to introduce plot points. The rise of an indie roleplaying game scene a few years later led to a profusion of games experimenting with removing the gamemaster, including *Capes* (2005), *Mythic RPG* (2006), and *Polaris* (2005; discussed later in this chapter).

Designer Jason Morningstar is also well-known for his games championing GM-less play, such as *The Shab-al-Hiri Roach* (2006) and *Fiasco* (2009). Morningstar prefers to say his games “distribute authority” differently than in traditional RPGs, noting that this creates a number of unique challenges for designers: “Is authority evenly or unevenly divided? Is authority isolated, binary or universal when invoked? Is authority sequenced or is it a constant?” (qtd. in Appelcline, 2014, p. 295) These questions would not even come up for the designer of a gamemastered game.

functions. Rather than provide a setting or other tools to help spark creative ideas, *Microscope* has no default setting or inspirational art and almost no supplied creative hooks for players to build on.¹⁷ *Microscope*, like Plotkin’s game, might be called a minimalist collaborative storygame, and one of the ways that minimalism takes form is by offloading and scaffolding the generation of ideas almost entirely from the system to the players.

The rules also note that while this creative authority is shared, it is never ceded. Two adjacent section headers read “You Build on Each Other...” “...But Don’t Collaborate”:

Nothing will kill your game faster than playing by committee. When it’s someone else’s turn, don’t coach... don’t suggest ideas. Even if another player wants ideas, don’t give them. Let them come up with something.

... If you collaborate and discuss ideas as a group, you’ll get a very smooth and very boring history. But if you wait and let people come up with their own ideas they may take the history in surprising and fascinating directions.
(p. 27)

The rules elsewhere (p. 73) explain that this is in part to ensure each player has an equal say in creating the world, rather than allowing more socially aggressive participants to dominate. So *Microscope* has equally shared creative authority not just as an aesthetic, but as an integral part of the game built directly into its mechanics:

I talk a lot about how *Microscope* forbids collaboration or brainstorming, but that’s not really true. What it does is require that collaboration happen through the medium of the game, rather than through open discussion and normal social rules. ... The entire game is a dialog, just a dialog with its own rules. (p. 75)

¹⁷The original game provides only a single page of one-sentence setting ideas, such as “A race of machines unearth their organic origins” (p. 58). A later companion volume, *Microscope Explorer* (Robbins, 2015), featured a more extensive set of ideas, but even this book’s page-long prompts are a far cry from the pages and pages of worldbuilding provided in a traditional RPG sourcebook.

This works in part because nothing is ever destroyed in *Microscope*: the rules point out that if one player “nukes Atlantis”—some part of the history that other players enjoy—those players can spend the rest of the game, if they wish, continuing to explore that element: everything that happened before its destruction, or the story of what its survivors did afterward. The rules elsewhere describe this as “great power without great responsibility.” Because there is no simulation of an ongoing world with consistent characters controlled by specific players, the creative authority given to the players cannot be used to interfere with what other players can and cannot do: “power over the fiction doesn’t have the same relationship [in *Microscope*] to power over play” (p. 72).

The lack of a simulated world is another interesting feature. Players are constantly creating, but those creations are static: other players may make events that happen before or after them, or even recontextualize them, but may not interact with the events themselves. It’s as if each player can add things to the Wunderkammer, but no one can change what’s already there or take anything away. Because of this, *Microscope* needs no simulationist rules for movement, combat, skill checks or random events: there is no *now* with a corresponding need to determine what logically should happen *next*, and there is never a question of whether a player’s intended contribution to the fictional world will be successful or not. As a result, *Microscope* needs no dice, no statistics, and no tables of data, and its rules are concerned almost entirely with helping players make their history compelling.

For example, one of the additional mechanics states that each time all players have finished taking a turn, the next player in sequence picks a “Legacy,” something

that happened in the last round that they feel should stay significant to the ongoing story. That player can then take a turn exploring any of the Legacies created so far. This mechanic provides common touchstones that are returned to again and again as the telling of the history unfolds. Several other mechanics exist to tweak *Microscope*'s play to encourage more coherent stories and better enable players to create them.

Microscope, then, explores a fascinating corner in the space of radically collaborative storygames. It demonstrates the Gesamtkunstwerk philosophy that these games are not necessarily best understood as descendants of earlier media, but as holistic creations with their own unique affordances and limitations. It also shows that qualities such as simulationism, linear storytelling, and unequal divisions of creative authority are not inherent to this mode of storytelling, and that the horizons of this genre are broader than commonly perceived. This excitement can be seen in many of the critical reactions to the game:

[*Microscope* is] a unique storytelling engine that sweeps away blinders of limits we enforce on the medium, which, I hope, will help us better realize the full potential of this form. There is so much more we could be doing. *Microscope* is a great start. (Torner, 2012)

In the terms of our framework, *Microscope* provides a procedure for Administration that enables players to Generate and Storywright more effectively, in part by deliberately not making any Generation contributions of its own. It also deliberately eschews Negotiation as a way of encouraging more egalitarian Storywrighting to arise from all participants equally. The next game we'll consider takes a different tack, centering Negotiation as the prime way that different ideas about Storywrighting are resolved.

7.3.3 *Polaris*

Another indie roleplaying game without a gamemaster, *Polaris* (Lehman, 2005) gives players control of individual characters, as in most tabletop roleplaying games, but imposes upon them a predetermined tragic narrative arc. Players negotiate with each other over the particulars of how that arc plays out, with each player fighting for their character and against a designated antagonist whose role is to ensure that tragedy eventually arrives. Unlike *Microscope*, *Polaris* breaks from the expansively permissive “yes, and” approach of improv theater, turning the discussion of possible futures into a mechanic based around sacrifice and difficult choices.

Players in *Polaris* create characters who are knights defending a dying civilization, a “snowflake melting beneath a demonic sun.” Each knight is destined to fall into corruption as they battle with the demons slowly destroying their kingdom from within and without. This is made clear in the setting materials for the game, but also encoded explicitly into the rules. Each character begins (along with a small number of other stats) with a Zeal score of 4. As characters face impossible fights and make heartbreaking sacrifices, they gain Experience, which has a chance to reduce Zeal. When a character runs out of Zeal, they replace it with a Weariness score. Whenever a character with Weariness gains Experience, their player may choose to narrate their character’s tragic fate, succumbing to demonic corruption—and must do so once Weariness reaches 4. In this way the game encodes an arc of tragedy into its mechanics. While some have speculated that a tragedy like *Hamlet* would not work in an interactive medium

(Bernstein and Greco, 2004, p. 178), *Polaris* provides a working counterexample. Players fight for their heroes, but with the knowledge that they are doomed to failure, and that the way that failure plays out will make for compelling drama.

While the rules of *Polaris* provide a fixed dramatic arc to go with its tragic worldbuilding, the moment-to-moment beats of drama are handled in lieu of a gamemaster by players with particular meta-roles. A player advances the story by staging a scene starring their character, who becomes the Heart for that scene. The player sitting opposite the Heart becomes the Mistaken, representing all opposing forces: demons, antagonists, and even the dark temptations within the Heart's own mind. The Mistaken's job in a scene is to try to make things as bleak as possible for the Heart, who fights for the survival of their character and their character's civilization.

In another game such conflicts might be resolved by dice or gamemaster fiat. Players of *Polaris* resolve conflicts through ritualized negotiation. The rules describe a set of key phrases that can be invoked during a scene to help determine the outcome. While there are twelve key phrases in total, the most important is "But Only If... ." A player can use this phrase to indicate that what the last player just said will happen only in the given conditions. This is primarily used by the Mistaken to complicate the Heart's actions, adding a consequence to what might have been an otherwise uncontested victory. The first player may then respond with another "But Only If...", continuing to raise the stakes; they may give up on both their original intention as well as the complication by instead saying "It shall not come to pass"; or they may accept the complication and end the scene by saying "And that was how it happened."

An example from the rulebook (p. 85), slightly condensed, effectively demonstrates this mechanic in practice. The current Heart, a knight called Mirzam (who has enough Weariness that his death or corruption is a possible outcome), is fighting the evil Solaris Knight on the highest tower of one of the remnants of the city:

Carrie (Mistaken): So, you're fighting back and forth . . . he brings his sword down on yours . . . [and it] shatters into a thousand pieces.

Paul (Heart): *But only if* the central shard of the sword drives through his chest, pinning him to the spot forever.

Carrie (Mistaken): *But only if* his blood seeps into the remnant, corrupting its people.

Paul (Heart): Whoah... *but only if* I die.

Emily (Full Moon)¹⁸: Wow... Are you sure?

Paul (Heart): *nods.*

Carrie (Mistaken): *And that was how it happened.*

Paul (Heart): Can I end the scene? *And so it was* that Mirzam died, his corpse slumped over the Solaris Knight that he had imprisoned, their blood mingling together and tainting his home, under a starless winter sky...

Rick (New Moon): Wow.

Paul instead could have said *It shall not come to pass*, in which case his sword would not shatter, the demon would not be pinned, and his character would not have died—to which his Mistaken would most likely have narrated the evil Solaris Knight escaping. Presumably Paul chooses his death as a form of bait to his Mistaken, a way of ensuring he gets what he wants—the death of his nemesis—even at terrible cost to both himself and his people.

While not all confrontations in *Polaris* are quite this dramatic, the mechanics enable a fascinating new kind of collaboration. The Mistaken's singular focus ensures that one player is constantly thinking about how to complicate the story for another—how to

¹⁸The additional roles of “New Moon” and “Full Moon” can suggest ideas and play supporting characters in a scene.

keep the tragedy on pace—while the key phrases drive players towards tough decisions between getting what they want at a cost or not achieving their goals.

Polaris also demonstrates how mechanics and setting can mutually reinforce each other to great effect, and how the particulars of a certain genre or style of story can be encoded into its mechanics. Newer tabletop storygames such as *Fiasco* (Morningstar, 2009) and *Lovecraftesque* (Fox and Annison, 2016) have continued this tradition; the latter, for instance, incorporates a number of rules designed to produce improvisational stories with the same kinds of dramatic structure as those by H.P. Lovecraft, such as limiting when violence and supernatural elements can appear in the narrative and when and how the protagonist may respond to them.

Polaris suggests that there are more ways to think about collaboration in storygames than are commonly recognized. The final game we’ll detail in this section keeps the role of gamemaster but makes story creation a continuous negotiation between gamemaster and players.

7.3.4 *Apocalypse World*

Release in 2010, D. Vincent Baker’s indie RPG *Apocalypse World* has since inspired dozens of spin-offs using its core mechanics (known separately as “Powered by the Apocalypse”). These spin-offs themselves have gone on to be influential and widely played, including *Monsterhearts* (2012), *The Warren* (2015), and *Dungeon World* (2015).

Powered by the Apocalypse has been widely praised in indie roleplaying circles¹⁹ for

¹⁹Games driven by the engine have won ENnie awards (given by the Gen Con roleplaying convention) in 2010 and 2013 and been nominated in multiple categories in several subsequent years.

streamlining the complexity of roleplaying rules into a series of “moves” that players can trigger through narrating their characters taking some action. For instance, a player in the fantasy variant *Dungeon World* describing their character leaping from a ledge to skewer an enemy with their sword might trigger the “Hack and Slash” move, which lays out potential outcomes for an action that involves an attack. A move is resolved with a dice roll, and can either succeed, succeed with consequences, or fail in an interesting way. This fiction-first philosophy²⁰ makes for fun, playable games, and is no doubt a big factor in the system’s success: the system prioritizes Generation of ideas over simulationist Administration, flipping the dynamic present in traditional rules-heavy games.

However, *Apocalypse World* and its variants do provide a unique form of Administration for the gamemaster, who also operates according to a series of moves. The moves specify the kinds of actions a gamemaster can take, and only allow them to make a move in particular situations—such as when a player misses a roll, or when the players look to the gamemaster to find out what happens next. The rulebook frames GM behavior as a particular procedure that should be followed to ensure a quality story arises: “There are a million ways to GM games; *Apocalypse World* calls for one way in particular” (p. 108).

The gamemaster’s moves encode ideas of notions of pacing, dramatic tension, and good storytelling in a proto-algorithm for Storywrighting. In much the same way

²⁰ “Powered by the Apocalypse” specifies that moves should always follow from the fiction: a player should say what they want to do in the story world, and only involve a move if it seems to match their intent. Contrast this with *D&D*’s “rules-first” philosophy, in which the example above might have been ruled out by the gamemaster because the player lacked a particular skill, because the combat rules don’t allow for movement and action in the same round, because falling damage might interrupt the players attack...

D&D helps a novice storyteller create compelling foes, *Apocalypse World* essentially provides a “do-it-yourself kit” for on-the-fly storytelling that remains dramatic but also responsive to players. This novel idea is worth examining in detail.

The rules lay out three tiers of behavior for gamemasters, starting with a high-level Agenda of three items:

1. Make the world seem real.
2. Make the player characters’ lives not boring.
3. Play to find out what happens.

The third item encodes the notion, frequently reinforced in the rules, that gamemasters should not prepare plots in advance. Their actions should be motivated by “being a fan of the characters” and wanting to find out what happens next in their stories:

Everything you say, you should do it to accomplish these three, and no other. It’s not, for instance, your agenda to make the players lose, or to deny them what they want, or to punish them, or to control them, or to get them through your pre-planned storyline (DO NOT pre-plan a storyline, and I’m not fucking around). . . . Go chasing after any of those, you’ll wind up with a boring game that makes *Apocalypse World* seem contrived, and you’ll be pre-deciding what happens by yourself, not playing to find out. (p. 108)

The rules flesh out the Agenda with a set of Principles, which are mostly either techniques for following the Agenda or procedural advice for enacting it. For example, “Name everyone, make everyone human” helps make the world seem real by ensuring there are no faceless goons in the story; “Address yourself to the characters, not the players” is a technique for keeping players engaged in the fiction.

The real meat of the game’s rules for GM behavior, however, are in the list of gamemaster Moves (Figure 7.6). The moves are each blueprints for plot events designed to raise tension, with either immediate consequences or the threat of future danger. The gamemaster does not literally say the name of the move (this is in fact one of the Principles) but uses it to inspire a fictionally appropriate development in the story. Many of the moves involve turning responsibility for advancing the plot and deciding consequences back on players: for instance, “put someone in a spot” might mean adding a new danger that forces the player to come up with a new strategy that addresses both problems.

The rules instruct the gamemaster to say “What do you do?” after every move. As in *Polaris*, this key phrase (which may never even be recognized by the players as such) causes a particular storytelling effect: putting the burden of continuing the story back on the players. Gamemastering *Apocalypse World* can feel a bit like being in a tennis court: continually seeking ways to put the storytelling ball back on the side of the players, while doing so gracefully and maintaining the rhythm of the volleys.

This philosophy also extends to the way the fictional world and the larger narrative are created. Again, the rules require (rather than merely encourage) that the gamemaster not prepare a story, or even a setting beyond a few basics, in advance of play. Gamemaster and players will design the world during the process of character creation, with all parties given power to invent details and fully collaborate (though the gamemaster serves as final adjudicator). A player who invents the detail that their character comes from the Wastelands might find the GM asking them “Oh—what’s

- Separate them
- Capture someone
- Put someone in a spot
- Trade harm for harm
- Announce off-screen badness
- Announce future badness
- Inflict harm
- Take away their stuff
- Activate their stuff's downside
- Make them pay resources
- Tell them the possible consequences and ask
- Offer an opportunity, with or without a cost
- Turn their move back on them

Figure 7.6: The GM moves in *Apocalypse World*.

it like there?” Asking questions is framed as the gamemaster’s primary role during character creation.

After the first session, the gamemaster invents adversaries (called “fronts”) for the players based around their interests, expertise, and weak points, and a plan for these villains to follow, assuming the players do nothing to stop them. The evolving story is then driven by the players as they react to the actions of their adversaries. A gamemaster might create an evocative setting in advance of play, but is generally encouraged to show up to a session without preparation, allowing the players to drive what happens.

Apocalypse World’s rulebook cites as inspiration the essay “Narrativism: Story Now” by Ron Edwards (2003), designer of the influential narrativist RPG *Sorcerer*. Edwards in turn adapts a framework from Lajos Egri’s *The Art of Dramatic Writing* (1960) to the study of story-centric roleplaying games. Egri describes a method of dramatic writing where a playwright comes up with a *premise*, a statement of morality which the play will make an argument to support. The central character, their conflict,

and the conclusion to the play will follow from the premise. Part of the audience's entertainment value arises from coming to understand the premise and appreciating the way the author makes a case to support it through the characters and their dilemmas. Edwards argues that the correct application of this framework to an interactive medium is to put all participants in the role of playwright. The premise of a narrativist roleplaying game is then not a statement, but a question, and each player, through their character, can make arguments for how that question should be answered:

Egri's statement-construction is very useful for the single author faced with a blank sheet of paper, with the goal at hand being a finished script. The audience will see the play, not the process of creation. However, in the role-playing medium, not only are there multiple authors, but the audience is also composed of these same authors, and their appreciation of the material occurs simultaneously with the significant creative decisions. Therefore, the Premise's imaginary resolution [the theme] is up for grabs among the group in role-playing, just as it is up for grabs within the author's own head before the play reaches final draft. . . . Think of [the theme] as the conclusive "uh!" that may accompany the climax and resolution of a story. It's uttered by the playwright as he hits a certain key or scribes a certain sentence, by the audience members at a certain point as they view the play, and by role-players *in both capacities* during the session, often simultaneously.

Therefore while a film about vampires might be driven by a premise that sustaining one's immortality through killing others inherently leads to moral corruption, a roleplaying game might instead let each player's character struggle with this dilemma, and come to conclusions about it, on their own—with other player characters providing dramatic foils and perhaps sources of conflict on the way.

Apocalypse World's rules turn the gamemaster into a facilitator for allowing players to explore the themes inherent in their characters. The gamemaster builds not a pre-determined story, but threats designed around the players' characters, using those

threats to force them into action. The limitations on gamemaster power shift control of the story to the players, while keeping someone in the role of facilitating (and at all times advancing) the narrative. The system also demonstrates how the Administration collaborative action can help with more than just the simulationist aspects of a game (such as combat), providing the scaffolding and glue to support a compelling emergent story as well.²¹

7.4 Bringing in the Digital

We have now looked at four analog collaborative storygames, and seen different ways a system can distribute the various collaborative roles between a set of players or even itself: some of these solutions have hinted at surprising revelations for the potential

²¹A more recent game, *Hillfolk* (2013), encodes an even more precise theory of dramatic storytelling into its rules. Constructed around character-driven drama, players pose scenes of external action driven by explicitly modeled internal emotional goals for their character. Specifically, the player's character takes the role of an *emotional petitioner* to another character, who will either grant or reject the petition. Based on how they advance and respond to petitions, both players and the gamemaster can earn and spend tokens for controlling aspects of the story, encouraging an economy where characters must first struggle with their emotional goals before advancing the external plot. *Hillfolk*'s designer devised this system after a thorough analysis of several classic plays and screenplays with the goal of operationalizing the process by which characters strive to achieve emotional goals, documented in his book *Hamlet's Hit Points* (2010).

A fascinating earlier example moves in the other direction of complexity. *Mythic RPG* (2007), among the first GM-less systems, included rules for a "Game Master Emulator," which is described as like "an artificial intelligence. It is designed to use simple rules of logic to answer any yes/no question" players have about the world. This somewhat oversells the system's narrative logics, which are simply that players decide how likely a question is, then roll on a weighted probability table for a yes or no response. The result is interpreted by all players based on whatever makes the most sense in the fiction so far. This notion that a gamemaster can be replaced by a coin toss is tempting in its simplicity, and echoes the theater game where a random audience member creates a story merely by asking yes or no questions to a supposedly expert storyteller. In reality, the storyteller simply answers according to some arbitrary rule, such as whether the question began with a vowel (Johnstone, 2012, p. 109; and see also my discussion in 8.6 of *The Prisoner in Block Nineteen* and its inspirations). The "Game Master Emulator" was even published as a standalone book (Pigeon, 2006), positing that all the rules in an RPG system could be replaced by its lone mechanic: "Do I hit it?" simply becomes a random response, weighted perhaps as "likely" if your character is a barbarian with an axe.

role of systems as collaborators. Let's now briefly look at a few digital storygames that flirt with ideas of collaboration. (Some of my own projects in this space are highlighted in chapter 8.)

7.4.1 *Elegy for a Dead World*

The promise of collaboratively writing a story with a game was at the heart of the 2014 game *Elegy for a Dead World*, in which the player explores alien environments and is invited to write text when they reach certain positions, which remains there and can be read again when that point is returned to. The writing mechanic encourages a deeper reflection on the player's exploration, inviting them to place it within a narrative context and connect unexplained elements together into something coherent. However, like my own *18 Cadence* (discussed in 3.3.2), *Elegy* does not attempt to understand or respond to the player's contribution, nor attempt to judge what makes a compelling or coherent story. But the affordance of inviting the player to make freeform creative contributions is still rare in games, and a necessary first step towards deeper forms of collaboration.

7.4.2 The Squinkifer Oeuvre

The games of Dietrich Squinkifer (Squinky) often mix real-world interactions between players with digital mediation, creating playful explorations of awkwardness, gender identity, and technology's role in mediating social interaction. Players are often invited to perform, enact, or interact with procedurally generated text, alone or together,



Figure 7.7: *Coffee: A Misunderstanding* in progress.

often making for unique and charming takes on the notion of system-assisted collaboration. In *Most Sincere Greetings, Esteemed One* (2016; created with Jess Marcotte), two players are given a procedurally generated greeting ritual and asked to enact it, often overcoming awkwardness around touch and public performance to do so. The generated nature of certain instructions (such as those involving body parts) creates a social tension around exactly how awkward any given generated instruction might be.

In the interactive play *Coffee: A Misunderstanding* (2014), audience members are invited up on stage to perform a scene where an internet fan meets a webcomic artist. Two additional participants sit behind each actor, making choices for them from a

conversation tree; the performers themselves do not see the available choices, merely the cues for their next lines. Technology is used in multiple ways in the piece. A screen behind the actors shows the audience upcoming choice points and which option is selected. The actors each receive their lines on a handheld phone, and the puppeteers also use phones to choose what their performers should say next. Finally, in some performances a band (also sometimes chosen from the audience) is asked to provide musical accompaniment based on cues selected by a master of ceremonies. *Coffee* uses technology to enable a group of strangers to awkwardly collaborate on a performance of a play about (among other things) awkwardness and collaborations, and is a unique demonstration of how many shapes collaborative storygames can take. Using the language of our framework, games like *Coffee* provide Administration around a rarely-addressed problem in multiplayer gaming (overcoming social anxiety), and invite players to Storywright their way together around the awkward seams between written and generated content, and between analog and digital spaces.

7.4.3 *Sleep is Death*

Jason Rohrer's 2010 game *Sleep is Death* provides a sandbox where one player can interact with a low-res game reminiscent of an early graphical adventure, moving figures around a screen and typing commands. Rather than being processed by a computer, however, these commands are processed by another player, who can react by changing the world in various ways: adding, changing, or removing objects to indicate state changes in the world, or having characters speak responses. This produces the



Figure 7.8: Screenshot from *Sleep is Death*, showing the “wizard” player’s interface for modifying the scene.

magical experience for the first player that they’re playing a game that can react to anything they might think to do—albeit with a lengthy delay to allow the human “wizard” to think up and instantiate a response.

Sleep is Death is another novel form of Administrative collaboration, providing a platform that allows two humans to have a unique kind of collaborative storygame experience. While much the same effect could be achieved by a gamemaster and a player roleplaying face to face, the digital veneer invokes a limited style of game and reinvigorates it by making it seem magical. While in my own experiences with the game I find it awkward to play as the storyteller (who must rapidly manipulate a low resolution interface to find and modify objects in the scene), Rohrer’s game is another

example of digital mediation enabling a new kind of collaborative game experience.

7.5 What does it mean to collaborate?

What role can systems hope to play in the collaborative storygames of the future? Can we make incremental progress that does not require achieving the dreams of the holodeck (see 5.1) or the nightmare of replacing human storytellers? To answer this, we need to look more closely at what precisely we mean by “collaboration,” to find out if the term can actually have useful meaning when speaking of an interaction between a human and a computer, algorithm, or system.

7.5.1 Three Questions

I would first like to show that collaboration is a meaningful term to use in this context by addressing three valid questions one might have about this assertion.

7.5.1.1 Why would we want to collaborate with a system?

We might rightfully ask why, in an age where people are increasingly turning to screens for entertainment, we would want to encroach on one of the remaining categories of games that remains largely technology free? There are a number of legitimate responses, including the previously made observation that technology has always been connected to these games. I’d like to focus on two answers to this question, however, both of which are about enabling new forms of expression: technology as a platform for collaboration and technology as a participant in collaboration.

Wardrip-Fruin (2005) has drawn a distinction between electronic literature where the reader mostly chooses between alternatives and that in which the reader can genuinely make creative contributions, framing the latter as *textual instruments*.²² When one “plays” such a work, as with a musical instrument, one is certainly limited by its affordances; but one may also be expressive, creating something genuinely new and surprising. Again like musical instruments, they may require skill to operate; but they may also include design features to help operators make a better performance, such as the frets on a guitar. Tabletop roleplaying has been compared to improvisational jazz (Edwards, 2003)²³ in the way participants constantly riff off each other within a particular framework to build something that delights and surprises them all. Technology can assist this collaborative play in new and surprising ways, providing new platforms for human-to-human collaboration or for individual humans to explore their own creativity.

Second, as augmented reality (AR) technology matures and becomes more widespread, a radical new design space opens up for games that take place between people in the real world but are also radically reshaped by technology.²⁴ While AR can provide a platform for human-to-human collaboration, it also has the potential to enable new kinds of collaboration between humans and systems representing fictional spaces, objects and characters. Beginning to formulate design frameworks and technological solutions now for ways in which advanced systems can collaborate with human participants will

²²This distinction mirrors in some ways the one made in 1.1.3 between games with and without expressive input.

²³See also the introduction to *Sorcerer* and its assertion that “the rules you hold are just some instruments. The music is up to you” (Edwards, 2001b, p. 10).

²⁴Virtual reality will also allow the same to be true for connecting people not in the same physical spaces; services like “Roll20” already allow geographically dispersed tabletop groups to play together, and this will only become more commonplace as technology allows for more and more realistic telepresence.

be crucial to exploring and humanizing this new medium. We will consider augmented reality in the terms of a “common ground” for system/human collaboration in (7.5.3).

7.5.1.2 Is “collaboration” a mismatched metaphor?

It’s also fair to ask whether I’m anthropomorphizing when I speak of collaborating with a system. When we use certain analogies to communicate a system’s behavior, we run the risk of an ill-fitting metaphor blinding us to the actual workings of the system in question (Agre, 1997; Kuipers et al., 1976). We might therefore consider whether some other metaphor fits better for our purposes, or if a systems-first approach eschewing metaphor entirely (such as that pursued by the discipline of software studies, Fuller 2008) would lead to better understanding.

We can define collaboration (and indeed the dictionary built in to my operating system does) as “the action of working with someone to produce or create something.” Collaboration requires *creation* of something (even if ephemeral); it requires *multiple parties* contributing to that creation; and it implies a *mutual respect*, such that contributions from any party will be at least considered by the others. Given these points, is it productive to consider a non-sentient entity, such as a piece of source code or block of rules, as a party to collaboration?

We might structure an argument for “no” around the notion of consent. The legal framework of most Western countries, for instance, assumes that no entities other than adult humans can consent to sexual intimacy. Even if a party not in this category (such as a teenager) claims they consent, they legally cannot because the law states only

adult humans have the maturity, right, and clarity of mind to do so. Is collaboration a form of activity that requires consent? Since computational systems have clearly not yet reached the sophistication of teenagers, is it premature to talk about collaboration with them?

This chain of reasoning overlooks two significant factors. The first is that humans and non-humans clearly engage in all manner of productive enterprises together, even though the latter can't offer informed consent; and in fact expanding these cross-species connections and striving to make them increasingly beneficial for all parties concerned is vital to our collective survival (Haraway, 2008, 2016). The second, following in part from the first, is that the way we treat others defines us, even when those others are not equals. One argument for vegetarianism is based not on empathy for animals, but on the grounds that the way we treat animals impacts the way we treat other human beings: an atrocity where humans are treated "like cattle" becomes less conceivable in a culture where cattle are treated humanely. We might therefore learn (or say) something about ourselves from the way we collaborate with other kinds of entities. My view, therefore, is that non-sentients are both qualified as potential collaborative partners, and in a position to teach us more about how we collaborate with each other.

As an example, consider Emily Short's *The Annals of the Parrigues* (2015a), a non-interactive travelogue of a strange fictional country. The text was created by both the human author and a generative text system she created, in an iterative process. A locale's description would first be generated, and the author would then modify the generator to make improvements before having it generate the next locale. Short

writes about wanting the procedurality and the act of collaboration²⁵ to remain visible, deliberately leaving in some generation artifacts that seemed artificial:

This is why the system will occasionally come up with character names like *Maria the Material*, or have a character buy a polyester codpiece, or put up a pub called *Sign of the Electoral Stallion*. It would be possible to smooth out what is uneven here, but I choose not to. It is the fingerprint of my collaborator. (p. 84)

This seems an intriguing lens to apply to human/system interactions in the context of creative content production, but does this stretch the collaborative metaphor too thin? Could we speak of a paintbrush as a collaborator, or any of the other storygames discussed in this dissertation (such as *Adventure* or *Prom Week*)?

Let me posit two rough metrics for when the lens of collaboration is useful to apply to human/system interactions. First, collaboration is a discussion about what is possible. A paintbrush can be

Ravencester Way *Relaxation — Veils of Blossom*

The road to Ravencester Way passes through orchards of tangerines, which are especially lovely in the spring, when covered with flowers.

Where the main streets meet, one may find a small sign warning against handouts to beggars.

Lodging Those in need of a bed may find NIGHTINGALE AND PARSELEY in the jewelry district. Emerald light pours in through the stained windows. Should the innkeeper request payment in copper coin rather than silver, custom holds that she is a were-stallion. In this eventuality, sprinkle the threshold of the room with basil and do not cross the line of protection until moonset.⁹ On the topic of travel preparations, it is also wisest not to venture into this area without a portable sunshade, especially if one comes from more northerly climes. A traveler's eyes may be sensitive to glare and their skin may grow red with excessive exposure.

Residency Taking up residency in this place is a difficult proposition, as the citizens do not welcome immigrants. Marrying one of the citizens may be the easiest method of obtaining general acceptance.



Bredark *Birdsong — New Spring Leaves*

Bredark was never founded, but has been inhabited continuously from the Time of Giants on. Another, grander people occupied this land once upon a time, and their buildings and towers can still be seen shattered across the landscape.

⁹ The credulity of the inhabitants of the town renders their conversation particularly charming, if surprising, to the visitor.

Figure 7.9: A page from *Annals of the Parriques*.

²⁵Short also writes about her digital collaborator being capable of surprising her, recounting what she considered its first joke in its description of the meals presented to the narrator at an inn: “Once we were served hound in a nut crust; and goat in a nut crust is not without its merits. The mare in a nut crust was less satisfactory.” The system had selected three random dishes, but the area in question had extremely limited cuisine: “I had not anticipated this blighting indictment of the culinary sameness of the town in question.” (p. 99)

interpreted as a symbol in such a discussion, or be used to enact its results, but cannot itself encode structured meaning in the semiotic way a book encoding an abstract system can. Second, even if such a system is present, it can only be a partner to collaboration by providing a foundation for it. The rulebook for the board game *RISK* (1959) expresses ideas about play but provides no structure for participants to provide their own: it allows for only one valid form of participation. Most classic adventure games are not truly collaborative for precisely the same reason: their structure invites only one possible response to each scenario (the correct solution to the puzzle), which is to their detriment when they present an illusion that such is not the case. *Apocalypse World*'s rules, by contrast, are almost entirely about responding to and integrating player contributions into the fictional world. Simulations like *Prom Week* might also perhaps be usefully viewed as collaborative partners (the notion is not exclusive with other aspects of a system). The key is that collaboration provides a platform for both expressive input and expressive response. While few systems do this effectively, I maintain the metaphor of collaboration can be useful even when not all parties involved are sentient, living partners.

Short's work on *Parrigues* is an intriguing example of treating a digital collaborator with respect, one of the implications in the word's definition. My own project *Almost Goodbye* (discussed in 8.5) experimented in a different way with incorporating text from a digital collaborator, asking a text generation system to narrate a specific category of sentences (those re-establishing context or controlling pacing) in an otherwise hand-authored story. I will next consider various ways we might productively view such

systems as collaborators.

7.5.1.3 What kinds of human/digital collaboration are possible?

If we return to the four roles we laid out in our design framework (Generation, Storywrighting, Negotiation, and Administration), and consider a non-sentient system making contributions during play in each of these categories, we might ask in which categories a system is most likely to make contributions that will be respected and potentially incorporated by collaborators. To answer this we might observe that the capacity for *understanding* contributions made by other parties is required for some of these categories but not others. Specifically, Ideas and Administration do not necessarily require understanding your partners, but Storywrighting and Negotiation do. This does not mean that systems cannot contribute to the latter two categories: but their manner of collaboration in those forms will be necessarily different.

We can thus discuss the collaborative contributions of a system in the context of two broad categories: Naïve Collaboration which does not require understanding, and Deep Collaboration, which does.

7.5.2 Naïve Collaboration

Consider the system of *Dungeons & Dragons*—the text in the rulebook, including rules, tabular data, worldbuilding text and even pieces of short fiction—as comprising the system with which human roleplayers collaborate to create a story. The human players (the gamemaster and those playing heroes) have slightly different responsibilities

from each other, but can we usefully consider both as collaborating with the system encoded by the rulebooks?

We can say that the *D&D* books act as a form of Generation, with ideas for what kind of story might be told at all levels from the game’s title on down.²⁶ The list of available character classes, the set of abilities those classes have, the inventory of items available for purchase, the lists of deities and monsters: all these are potential contributions to the story—although the gamemaster generally has more say than any other party about which of these suggestions get incorporated.

But the rulebook’s contribution is deeper than this. The rules and fiction reinforce each other to tell collaborators *what kinds of stories are likely to be most rewarding* for play. If a “Table of Falling Damage” exists in the rulebook, it tells us that falling from heights is something that could potentially happen in the story; that (based on the damage dealt) if present it is likely

²⁶We might think of this instead as a form of collaboration with the *designers* of the game, whose ideas about compelling stories were encoded by the system. But the designers are not there at the time of play. What the players are interfacing with is the *system* the designers created to champion those ideas. We also cannot (always) interrogate the designers to find out what their intentions were—but we can study the text of the rulebook, in a line of thought echoing literary reader-response theory (Tompkins, 1980, p. xv).

CHANCE OF CONTRACTING DISEASE	
Base Chance	2%
Modifiers:	
currently diseased or infested with parasites	+1%
crowding (city, encampment, shipboard)	+1%
filth (city, encampment, siege)	+1%
character is old	+2%
environment (marsh, swamp, jungle)	+2%
hot and moist climate (season or region)	+2%
character is venerable	+5%
exposure to carrier of communicable disease	+10%
cool weather or climate (high elevation, etc.)	-1%
cold weather, high mountains	-2%
shipboard after being at sea 2 weeks	-2%
Procedure: Adjust base chance of disease being contracted by applying modifiers. When the final percent chance to contract a disease is found, roll for each character concerned. If disease is indicated, go to the DISEASE TABLE below to determine what sort is contracted.	
CHANCE OF PARASITIC INFESTATION	
Base Chance	3%
Modifiers:	
filth (garbage, manure, sewage, etc.)	+1%
improperly cooked meat	+2%
polluted water	+5%
swamp or jungle environment	+5%
cool weather or climate, desert climate	-1%
cold weather, high mountains, cool desert climate	-1%
Procedure: As for disease, and check PARASITIC INFESTATION TABLE .	

Figure 7.10: A table from 1st edition *Dungeons & Dragons*, really less about probability than Generation of interesting ideas.

to be a serious source of tension; that regardless of being set in a fantasy world the story is still one where gravity exists, reckless actions have consequences, and sudden death is possible; and so on. *Polaris* provides a more obvious example of this, where the rules and worldbuilding combine to contribute a specific story of tragic downfall. But this form of Generation as a kind of collaboration—the idea, but also the potential consequences of incorporating it—exists even in more straightforward roleplaying systems like *D&D*.

Systems can offer another form of Generation-based collaboration by embedding other systems within themselves, such as a system for generating a random treasure, NPC, or entire world. The roleplaying sourcebook *Vornheim* (Zak S, 2011) compiles a number of procedures for unusual forms of on-the-fly generation that produce offbeat or unexpected results, such as using the shape of the numbers in a die roll to create a street map of an unexplored city district. By being unexpected, mechanically delightful, or deliberately weird, these contributions encourage human collaborators to take them up and Storywright them into the ongoing narrative, reinforced by *Vornheim*'s fictional setting of a bizarre and unpredictable fantasy city.

A system need not cede power over choosing which of its contributions to make part of the story. *JUGGERNAUT* (Morningstar, 2015) is a live-action roleplaying game where the player characters are locked in a room to test a computer that can predict the future. At certain times a player can draw the top card from a shuffled deck of predictions that are often highly narratively charged (such as “Within the next five minutes, two members of the test committee will reveal a relationship outside of their professional obligations”). The rules state that it is up to the players, without breaking

character, to ensure these predictions come true. Here the system's contributions *must* become part of the narrative. This is also true in a less direct fashion of many games that involves drawing immediately played cards or rolling on a table to find a result—but *JUGGERNAUT* has no human gamemaster to mediate these results, and the fiction and system work together to deliberately frustrate attempts by the players to Storywright what's happening. The result is a bleak emergent story reminiscent of a *Twilight Zone* episode, where the player characters struggle fruitlessly to explain the computer's ultimately meaningless but relentlessly accurate predictions. Here the system does not attempt to understand or respond to your contributions because that is part of the fiction.

The other straightforward way systems can collaborate is through Administration. We can frame any set of rules as a form of administrative collaboration, but more interesting are those that assist human players in carrying out the other activities. In Plotkin's journal game, the mechanics of the black and white stones give humans permission and parameters for Storywrighting their contributions together. The key phrases in *Polaris* provide structure to a style of Negotiation that reinforces the fiction of a descent into tragedy. The system in Squinky's multiplayer games acts as facilitator, helping human collaborators overcome shyness and awkwardness to have fun and sometimes moving experiences together. And *Apocalypse World*'s GM moves are a storygaming mentor in a box: a set of procedures designed to allow even a novice to help tell an emergent story that reacts to player ideas without railroading them into a predetermined plot. Dismissing Administration as a form of collaboration brushes

aside the very real contribution that even simple systems can make to a collaborative storygame, and prevents us from thinking about how more complex procedures and systems, including computational ones, can become meaningful participants.

7.5.3 Deep Collaboration

For a system to Storywright or Negotiate itself, however, requires overcoming two major challenges: *understanding* a collaborator’s contributions so far—what they have placed in the Wunderkammer and why—and *contributing* something that follows. This contribution is not quite the same thing as Generation, because here it comes explicitly as an act of conversation: following on from and continuing the ideas currently in play.²⁷ This is radically different from less contextual forms of generation.

How do humans understand each other in this sense? A study coding player speech during sessions of tabletop roleplaying games found a wide variety of discourse acts were used, including maintaining the social order, direct performance as characters, suggesting possible actions, acknowledgments and confirmations: but as much as 40% of communication was connected to “maintaining a shared understanding of a purely imagined fictional world [which] can be somewhat challenging, and involves the risk of communication problems. If the joint understanding of the imagined reality is not aligned closely enough to allow the players to interact in and with it, conflicts of understanding can arise” (Drachen and Smith, 2008). For example, a player whose thief is sneaking into a bedroom might need to ask if there’s a dresser and how far away it is before

²⁷ Another way of putting this is that we could say generation of ideas happens across all four activities, but only in the case of Storywrighting and Negotiation must it be contextual.

stating their intention to sneak over and pilfer it; a gamemaster who describes the thief stepping on a creaky floorboard might have to stop and negotiate with an indignant player who assumed the floor was stone. So a computer collaborator faces at least two difficult challenges in understanding: maintaining (including creating and repairing) an understanding of an imagined world, and updating it correctly as collaborators make new content (new lexia) within it.

To enable a computational system to understand and contribute implies overcoming both serious technical challenges and entering a design space that has rarely been explored. On the technical front, we would seem to need to solve problems of natural language understanding and processing; of model representation; of common-sense reasoning and domain-specific knowledge (such as the genre conventions that make, for instance, a bat significant in a vampire story); of social norms and cues (such as knowing when a contribution would be fun or appropriate for a particular group of players); of story generation, possibly involving goal-oriented action planning, computational models of narrative, and stylistically appropriate language generation; perhaps of player modeling; and, not the least, linking all these things together into a single framework. While progress has been made on all these fronts individually, this last challenge remains among the biggest and least-solved: a “lack of consolidation” in the field means wheels are continually reinvented and rarely interoperable (Arinbjarnar et al., 2009).

Lacking a unified technical framework to full understanding and contribution by a system, one approach might be to explore whether extremely limited forms of either are possible. For instance, we might imagine a game where a computer plays a character

who is physically separated from the others and can communicate through only a narrow channel, fictionally explaining the lack of a mental model of the fictional world. We can indeed imagine such scenarios—a voice through a portal from another world, a god or force of nature unconcerned with the particulars of the players’ situation²⁸—but we place severe limitations on the kinds of fictions we can tell via this approach. Alternatively, we could imagine reversing this: instead of teaching the system an extremely limited form of communicating with humans, we could teach humans the language of the system. Some projects have attempted to create user-friendly interfaces for creating a story graph that a system can reason about (Skorupski et al., 2007; Pizzi and Cavazza, 2008; Koenitz, 2011). However, it’s hard to imagine humans stopping their real-time play to encode even the simplest story graph into a computer. Even the simplistic story representation of *Sleep is Death*, for instance, is time-consuming and difficult to operate.

An alternative to dumbing down either participant’s native tongue so the other can understand it is to look for a third form of communication that is native to neither but navigable by both—a shared language that both humans and systems can employ. In *Computers as Theatre*, Brenda Laurel (2013, p. 11) explores a number of metaphors to help understand human/computer interactions. One of these, adapted from (Clark et al., 1991), is the notion of a “common ground” to represent how systems and humans communicate, which “forms a shared context for action in which both [human and computer] are agents.” The metaphor of “drag and drop” with a mouse pointer, for instance, is native to neither humans nor computers, but both can be taught to navigate

²⁸ *JUGGERNAUT*’s capricious computer is an example of the latter. Regarding the former, see also my minimalist take on this, *The Prisoner in Block Nineteen* (discussed in 8.6).

it. A system can be given a set of operational logics encoding the notion that a particular sequence of mouse events represents a command to move a file, and humans can accept and internalize the concept that moving icons on a screen corresponds to making changes to virtual data: “playing along” with the metaphor. This common ground allows for successful communication between humans and computers.

We do not need to teach computers everything about human communication for them to be successful collaborative partners. Can we imagine a common ground for communication about stories and characters? This question reframes our overwhelming technical challenges as a more limited problem encompassing both design and technical thinking. In the final section of the chapter, I will discuss methods for finding a limited but productive common ground where both systems and humans can pursue narrative collaborations.

7.6 Futures for Digital Collaborative Storygames

Coming full circle, what have we learned from studying analog collaborative storytelling games (and the digital hybrids making tentative steps into this space) that speak to a future for digital collaborative storygames?

7.6.1 Systematizing Naïve Collaboration

We have demonstrated that even simple forms of collaboration such as Generation and Administration can be surprising and useful contributors to an interactive story. *Apocalypse World*’s rules for gamemasters, for instance, are a form of high-level

pseudo-code for managing an ongoing story. While these rules still require a great deal of human common-sense and domain-specific knowledge to instantiate in practice, we might imagine formally proceduralizing these rules into a digital program that can act in a limited, particular story domain.²⁹ The historical centering of the “dungeon” as a locale for roleplaying, for instance, has been traced to the balance between freedom and constraint a contained but explorable environment offers, allowing gamemasters (especially in the formative days of the genre) a way to prepare a playable but bounded story (Henry, 2003).³⁰ We can imagine similar types of constrained settings making it more tractable for computers to be taught how to enact rules like those in *Apocalypse World* for helping administrate scenarios.

Apocalypse World’s rules are also constrained in a design dimension. In contrast to systems that attempt to model complete theories of narrative or generalized goal-oriented action planning, Powered by the Apocalypse is razor-honed for the particular problems faced by human players contributing to a storygame. Ironically, this streamlining and simplification was done to create a system that humans could easily handle at runtime, but the result is a system much easier to teach to a computer than the more complex and complete bases for narrative understanding mentioned above.

²⁹There has been some limited technical exploration in this domain already: see for instance (Peinado and Gervás, 2004). However, none of this work to my knowledge has yet been released in the form of a playable system.

³⁰We see this constrained freedom echoed in the room-based compass navigation of interactive fiction, the circumscribed island settings of *Myst* and its successors, and the delineated options in conversation trees.

7.6.2 Exploring common ground

Exploring productive common grounds that both humans and computers can use to collaborate on stories is a promising direction for future research. I previously (5.3.4.3) discussed Chris Crawford’s approach of creating a symbolic language that both humans and computer-controlled characters can use to communicate. Is there some equivalent to this for communicating about story, or emotion, or fictional spaces?

We might think of the themes in *The Ice-Bound Concordance* as a simple common ground allowing the system and the players to make statements about the shared story space. Each partner has simple affordances for strengthening a theme. *Ice-Bound* does not make this form of communication explicit, disguising it behind other game mechanics, but we could imagine extending it to push forward the extent the collaboration, an idea explored more fully in the following chapter (8.4).

Another potentially fruitful common ground is enabled by the emerging sophistication of augmented reality technology. Microsoft’s HoloLens (released only in prototype form at the time of this writing) uses sophisticated visual processing to gain a complex understanding of the physical space around the wearer, the position of their body and hands, and the relationship between both tracked physical objects and simulated virtual objects within that space. This technological breakthrough provides a richly complex new platform that both humans and systems can understand and manipulate, a shared language of motion, position, and dimensions—a literal Wunderkammer in which both humans and system can position objects both physical and virtual. We will need to

develop new design frameworks to take advantage of this platform. What new kinds of stories are enabled when both humans and computers can speak and manipulate the language of objects? My 2011 project *what if im the bad guy* explored positioning and repositioning fragments of a true story in physical space;³¹ *18 Cadence* was also concerned with the positional physicality of story fragments and their relationship to each other. Emerging platforms will doubtless allow these ideas to be taken much further.

Sculptural Fiction as common language. In chapter 3 I described sculptural fiction as a storygame framework focusing on imposing a structure on a set of existing lexia as the primary narrative mechanic, rather than navigating an existing map of nodes. Sculptural fiction approaches but does not reach collaboration as we’ve defined it in this chapter, since it features neither players nor system creating new nodes and neither is there inherently any mechanism for encouraging coherent stories. What sculptural fiction does potentially provide, however, is a foundation for bridging the communication gap that prevents a system from understanding its collaborators. If both the system and humans can be taught enough about the available lexia to have a working understanding of how they fit together to become a story, a space is opened up where both parties can make suggestions, understand each others’ contributions, and work together towards creating something they both are happy with.

³¹I have written about this in detail in (Reed, 2013).

7.6.3 Deeper Generation

While even simple generation can be an effective form of collaboration, there are also more complex techniques for contributing ideas.

The field of computational creativity explores ways systems can generate ideas that have more of the qualities we associated with human creativity. For instance, the AhaNet project (McCaffrey and Spector, 2011) models a set of abstractions about physical objects³² along with how often humans associate each with that object, allowing the system to look for uncommon but not obscure features that might be modified to form a creative modification and generate a creative variant. The system created a product idea (later taken to market by the designer) for a candle that sits on one side of a scale, snuffing itself out when it is nearly burned out. The system “discovered” this innovation by combining two typical facts about burning candles rarely considered by humans as significant, “changing weight” and “motionlessness,” and posited that the first quality could alter the second as a creative solution to coming up with a novel candle design.

There has also been recent promising work in systems that are more aware of the significance of their own generation. The Productionist engine (Ryan et al., 2016) expands on the idea of a context-free expansion grammar to allow for tagging possible expansions with metadata, which can be any form of information a designer finds significant. This allows for both *searching* for an expansion that meets a specific set of criteria and *recording* all of the tags returned in a particular expansion. This enables communication

³²The system can also handle modeling intangible domains, such as problem definitions.

between the grammar and some outside entity (such as a collaborator). Generated text can now understand what was communicated, and attempt to communicate particular things. Ryan and colleagues have used a version of this system in a project that remixed hand-authored *Prom Week* scenes in an attempt to dynamically make new scenes to narrate an arbitrary set of game states (Antoun et al., 2015). A related branch of work in mixed-initiative design tools places generation in the context of a particular design space, which both human and tool understand well enough to participate in (Smith et al., 2010).

Social Simulation as source of creativity. Likewise, we might view social simulation as a means of addressing our other problem with digital collaborative storygames: making creative and contextually appropriate contributions. Imagine a game like *Prom Week* where the world and social state are simulated by the game, but instead of performances realizing actions, players roleplay these scenes out. The input back into the system is whether the action was accepted or rejected. In the same way that *Prom Week* doesn't much care about the narrative specifics of its instantiations, only that they serve to narrate a specific system change, here the players would determine and perform the change, and report the results back to the system. This would be a more advanced kind of Administration and Ideas contribution (showing a huge range of knock-on social effects from player actions).

As another example, imagine a tabletop game where the gamemaster controls a town on their computer, as in "Talk of the Town" (Ryan et al., 2016; the system

behind *Bad News*) and inputs major player actions into it. The system informs the GM in turn of major events happening, as a result of these actions or otherwise, and can be queried to answer questions about what a particular resident thinks of another or a player, what the mood is like in a particular neighborhood, and so on, in real time. While this kind of detail would be difficult for a human GM to manage during play, a system can easily handle even complex simulations very quickly, freeing the GM up for performances, Negotiation, or other kinds of collaborative activities which humans are best at. Giving tabletop designers access to a tool like “Talk of the Town” opens up a space for a new kind of collaboration.

Design innovations for collaborative storygames need not wait for the hard problems of interactive drama to be completely solved. By considering insights from the world of tabletop storytelling games, which can ignore technical obstacles, we point towards a large unexplored space of digital games that can use existing or emerging technologies to create new kinds of play experiences.

In this chapter, I identified the space of collaborative storygames, connecting games from a variety of traditions and backgrounds via a shared focus on collaborative creation and systems for encouraging a compelling story. While tabletop roleplaying theory has previously worked at characterizing a subset of this space, a broader view focusing on the narrative logics that can enable this kind of play proves more useful. I do, however, find Lars Konzack’s Wunderkammer-Gesamtkunstwerk Model a useful starting point, particularly through his metaphor of the Wunderkammer as a shared

mental space to which all participants can add and help interpret story content. From this starting point I identify four primary activities that participants in a collaborative storygame engage in: Generation, Storywrighting, Negotiation, and Administration. Through analysis of a number of both analog and digital games in this space, I posit that non-human systems can be true collaborators across any of these categories of action, though Negotiation and especially Storywrighting are most difficult for systems to participate in (ironically, since much interactive drama research has focused on Storywrighting-like activities). I conclude by suggesting that this broader framework for narrative collaboration, informed by deep analysis of the narrative logics of both analog and digital collaborative storygames, points the ways towards new types of digital games that embrace different aspects of collaboration, and that this space is broader than has been previously suggested.

In the final case study chapter to follow I will detail some of my own experiments in exploring these spaces, including both analog and digital games.

Chapter 8

Collaborative Storygames Case Studies

While I have not yet produced a major work stemming directly from collaborative fiction principles, I have worked on a number of small experiments that have helped shape my thinking about this design space, and sometimes revealed useful insights. Each of these projects is summarized briefly below.

8.1 *Problem Planets*

In this 2013 experimental prototype with Brandon Tearse and Peter Mawhorter,¹ we attempted to use the Skald story generation system (Tearse et al., 2014) to create stories that a player could interact with via generated choice points. Skald is a reconstruction of the unreleased but often-cited MINSTREL story generation system (Turner, 1993) which had survived only in the description of its behavior in Turner’s dissertation. Tearse had first reimplemented this system in a project called MINSTREL Remixed

¹*Problem Planets* was part of a larger research project to prototype technology for interactive scenario generation, which has the goal of generating both gameplay and narrative together.

(Tearse et al., 2010) and then built on and extended it to create Skald. While Skald had been previously used in prototypes, Problem Planets was an attempt to use it to drive a complete game that could generate original stories in reaction to the player’s choices.

MINSTREL and Skald both work on the principle of “imaginative recall,” the notion of constructing new stories by modifying existing ones in an attempt to produce something that seems original but still coherent. To achieve this, the system needs a library of “stories” that are carefully constructed so as to be understandable to the case-based reasoning engine that drives it.² While such systems can in theory produce novel but sensible stories, they can just as easily produce stories that vary along axes that humans do not find sensible or interesting: and adding additional stories to help restrict this only introduces more elements which the system does not understand: “The nature of this adaptation is, precisely, that it exceeds the bounds of the knowledge already available about the microworld” (Wardrip-Fruin, 2009, p. 195).

Despite a significant amount of effort, our team was unsuccessful in achieving our goal of using Skald to drive a game based around interactive story and choice point generation. We had hoped that a domain in which simple, off-beat stories were part of the fiction³ would smooth out some of the system’s rough edges of the imaginative recall approach to story generation, using case-based reasoning to alter an existing corpus of stories to meet new constraints. In practice, however, this programmatic approach

²The scare quotes are because the artifacts that such a system can process must be so mechanically and logically constructed that some storytellers might raise an eyebrow at applying this label to the results.

³Problem Planets would have told the story of a pair of robots traveling the galaxy helping organic life forms solve problems, a nod to Stanislaw Lem’s famous robot characters, the constructors Trurl and Klapaucius. A mock-up of the game interface used a retro font and minimalist English in the hopes that this aesthetic would help generated story text seem less jarring.

failed to include the kind of coherence that humans expect even from very minimal stories. Notions such as cause and effect could only exist with careful planning that easily went astray, and despite laborious construction of example stories tagged with how each action and story entity were logically connected (see 8.1), the system would more often than not generate stories that were “legal” but incomprehensible.

In hindsight, and viewed through a collaborative storygames framework, *Problem Planets* was an attempt to make the system the Storywright: have humans Generate a large number of ideas (story fragments) and leave the computer to stitch them together. But this reverses the strengths and weaknesses of human and non-human collaborators. Expertly arranging a Wunderkammer involves a wealth of knowledge of all of the things outside it: what about the great world outside makes these particular objects and their juxtaposition so interesting, the “common sense” so difficult to teach to systems.

While many continue to pursue research in this vein and have made worthwhile incremental steps (see for instance Mawhorter, 2016), my view is that this particular hard problem will not be solved any time soon. In the interim we must look to other ways that technology might assist in collaborative storygames.

8.2 *SPUN Adventure*

In 2010 I released a simple extension for Inform 7 (a domain-specific language for parser interactive fiction) as an experiment in allowing players to bypass puzzles.⁴ The extension added the notion of “spin” as a binary state a player of a text game could

⁴The extension can be found at <http://lacunagame.blogspot.com/2010/01/spin.html>

```

/* A reporter tries to convince a doctor friend of some disturbing
   medical news. */

STORY REPORTER_INVESTIGATES .

REPORTER is a citizen . DOCTOR is a researcher .

RUMOR_BLUE_GOO_CAUSES_CANCER is a thought_info .
  PROOF_BLUE_GOO_CAUSES_CANCER is a thought_info. DNA_SEQUENCER is a
  scientific_instrument .

State1: DOCTOR has feelingstowards of negative for
  RUMOR_BLUE_GOO_CAUSES_CANCER .
State2: DOCTOR has feelingstowards of positive for REPORTER .

/* Scene 1: The reporter fails to persuade the doctor. */

Goal1: REPORTER wants DOCTOR to have feelingstowards of positive for
  RUMOR_BLUE_GOO_CAUSES_CANCER .
Action1: REPORTER persuades DOCTOR .
State3: DOCTOR has feelingstowards of neutral for
  RUMOR_BLUE_GOO_CAUSES_CANCER .

State2 motivates Goal1 .
Goal1 plans Action1 .
Action1 accidents State3 .
State1 hinders Action1 .

/* Scene 2: Because of their friendship, the doctor does some research
   and learns the truth. */

Goal2: DOCTOR wants REPORTER to have feelingstowards of negative for
  RUMOR_BLUE_GOO_CAUSES_CANCER .
Action2: DOCTOR inforequests DNA_SEQUENCER .
State4: DOCTOR knows PROOF_BLUE_GOO_CAUSES_CANCER .
State5: DOCTOR has feelingstowards of positive for
  RUMOR_BLUE_GOO_CAUSES_CANCER .

State2 motivates Goal2 .
State3 motivates Goal2 .
Goal2 plans Action2 .
Action2 accidents State4 .
Action2 accidents State5 .

```

Figure 8.1: Example *Problem Planets* story, in a domain-specific language created for authoring for Skald. Note that all English words here have no special meaning to the system: in particular, words like “motivates”, “plans” etc. are completely arbitrary labels. The theory is that the system will create stories with similar kinds of links in a way that continues to appear to be a sensible story, but we found this difficult to achieve in practice.

either have or not have. By solving a puzzle, a player could gain spin; if they were stuck, they could spend it to bypass a puzzle, via a hand-authored way appropriate to the story. I proposed (but did not actually create) a game called *SPUN Adventure*, adding spin to the classic Crowther-Woods game. For instance, if the player of *SPUN Adventure* could not find the key to the grate permitting access to the cave, they could spin the grate to trigger a narration they they found a nearby rock and used it to smash the lock.

SPUN Adventure would require the game writer to create custom text for each bypassable puzzle, but it can be seen through the framework developed in the previous chapter as a minimalist experiment in shifting narrative power in a digital game towards the player. Getting stuck on puzzles prevents many players from continuing a story or exploration they might otherwise be interested in. Were *SPUN Adventure* a real game, it would be only a small step towards true collaboration between a human and digital storyteller, but as a thought experiment it demonstrates, like Plotkin’s journal writing game, that a mechanic to shift narrative control need not be complex nor intelligent.

8.3 Minimalist Story Generator #1

This project⁵ generates short sentences that might be recombined to make tiny mystery stories. The sentences are of three types: motives (“Miss X stands to lose everything”), sinister actions (“She made careful arrangements”), and dramatic deaths (“Professor Q drowned in the lake!”). These are then animated moving quickly across

⁵Available online at <http://aaronareed.net/experiment-in-minimalist-interactive-narratives/>

You are in a 20-foot depression floored with bare dirt. Set into the dirt is a strong steel grate mounted in concrete. A dry streambed leads into the depression.

>UNLOCK GRATE

You don't have a key!

>SPIN

You have spin.

>SPIN GRATE

Casting around the streambed, you finally find a decent-sized rock, and bring it down forcefully on the lock. It smashes into pieces.

[You have lost spin.]

(later)

The snake rears up before you!

>SPIN SNAKE

You don't have spin.

>RELEASE BIRD

The little bird savagely attacks the snake, driving it away!

[Your score has gone up by 5 points.]

[You have regained spin.]

Figure 8.2: Hypothetical transcript from *SPUN Adventure*.

the screen. By clicking a sentence, the player can cause it to move into one of three slots at the bottom of the screen. The dynamic that emerges is a game of “catching” sentences that fit together to form a complete three-part story at the bottom of the screen, such as “The housekeeper was jealous. She bribed the chef. She smiled quietly.”

Here the computer takes the role of Generator, and the player the role of Storywright, with a game-like addition of quick reflexes and serendipity necessary to make a satisfying story. Again, this is by design a minimalist contribution: though this is barely a collaborative storygame by our definition, it shows in a small way a process via which both an algorithm and a human are contributing to produce a story.

8.4 *Ice-Bound* as Collaborative Storygame

We have already discussed *The Ice-Bound Concordance* as an example of sculptural fiction (Chapter 4), but we might briefly revisit a few of its features through the lens of collaborative storygames. A central mechanic of the game is combining existing lexia authored by the story creators, but despite it being a central theme of the story, to what extent is *Ice-Bound* truly collaborative?

To answer this, we need to look at the affordances both the system and the player have for selection of content. The game’s engine can both (1) select the lexia to build a level and (2) modify template text in that lexia’s long-form description. While neither of these methods are truly generative, they do help create the impression that the system is responding to player input. In both cases, the selection is in part a response

to the player’s prior contributions: their selection of themes to prioritize. However, *Ice-Bound* also (3) selects a theme of its own to emphasize each time the player does so, announcing its decision through the character of KRIS. While this decision seems largely random,⁶ it again helps give the impression that KRIS is making his own contributions to the story.

The player, in turn, has several affordances for selecting content. They (1) choose to activate or deactivate lexia to shape a particular level’s story, (2) choose an ending to each story that solidifies one possible theme (confirming which theme they meant later on), (3) choose a Compendium page perhaps in part to reveal an important plot point to KRIS, and (4) choose between variant text options in lexia that contain shimmering alternatives. The player, too, is merely selecting from available material, rather than generating anything new.

However, while the player’s input is constrained, the possible combinations mean it does approach expressivity. There are often more than a hundred possible configurations of lights for a particular story, meaning the player can genuinely explore combinations and hit upon insights of combinations of active symbols that weren’t immediately obvious to them. *Ice-Bound* also pays attention to this expressive input in a more elaborate way than many storygames do. Furthermore, the affordances that the player and the system have for affecting the story are quite similar: KRIS cannot select Compendium pages, but both he and the player are selecting symbol lexia, altering template text, and choosing themes to emphasize. These parallels hint at a “common

⁶In fact, KRIS’s selection of themes is influenced in part by which have the most remaining unread lexia, thus attempting to prevent the system from “running out” of content too early.

ground” which both parties are using to shape the story, and within which both can react to the others’ contributions.

What would make *Ice-Bound* more collaborative? My co-creator Jacob Garbe has begun work on a project⁷ suggesting one answer, in which the player can teach the system symbolic meanings for objects in (or qualities of) the game world—the notion that, for instance, a piano represents sadness, or darkness symbolizes safety. The player then navigates three-dimensional representations of conceptual spaces in which these signifiers are used to communicate the mental state of game characters. By manipulating these meaning-charged objects within strongly representational spaces, the player can understand and change the mental state of game characters. This evolves both the player’s input and the system’s response into a sort of action-based conversation about metaphor—rather than a fixed, one-way assignment of meaning, as in *Ice-Bound*’s themes.

There are many other ways in which we could imagine a framework like *Ice-Bound*’s evolving to represent more meaningful acts of collaboration. The system might make more extensive use of generative text to achieve more expressive output, perhaps through a searchable context grammar like the Productionist system (Ryan et al., 2016). This could enable much more of the game’s text to be responsive, communicating and adapting to both the system’s plans for the story and the player’s contributions. KRIS’s criteria for selecting and modifying lexia could become more elaborate, incorporating theories (or at least heuristics) for compelling storytelling, perhaps in the same fashion

⁷As of April 2017 codenamed *Changeling*, with a development blog at <http://project.jacobgarbe.com/>

that *Apocalypse World*'s rules help shape a better story (or in the even simpler way that Plotkin's journal game does the same). We could do much more to understand the player's input, looking at the way they've resolved variant texts as another window into their opinions about the story, or offering chances to make alterations to more axes than theme. It's clear that *Ice-Bound* is not an end point but a marker pointing the way towards further exploration, and that digital stories can become much more collaborative even without major technological advances.

As one final thought experiment, imagine a version of *Ice-Bound* where both the player and the system were able to create new, wholly original lexia. The hidden thematic tags in the current system would be made visible and alterable in this variation: as in *Elegy for a Dead World*, the system would not attempt to process the player's text, only considering the thematic tags attached to it. The system might use more dynamic expansion grammars to generate new lexia: still limited by the fragments provided by authors, but able to combine and remix them in more elaborate ways, to express the thematic notions each afford. Given this setup, both human and system could now make direct statements about how they would like to change the story—using the same shared language of fragments tagged with themes—and invoking additional fragments that follow on from those themes, as currently happens. This would move *Ice-Bound* closer to the realm of textual instrument, letting people create genuinely novel contributions to Kristopher Holmquist's endlessly variant text.

8.5 *Almost Goodbye*

Another experiment in minimal content generation, *Almost Goodbye* (2012) is a mostly hand-authored short science fiction story about a scientist’s last night on Earth.⁸ Before she leaves to board the colonization ship that will take her on a one-way voyage, she chooses which of the remaining people on her list to say goodbye to, and where—without enough time to get through them all. The experiment was to find a kind of sentence that could be turned over to procedural generation (using a text expansion grammar) that would produce output that I, as a human author, did not feel embarrassed by—that seamlessly fit in with my hand-authored prose.

I settled on generating “satellite sentences,” which moderate pacing and reestablish context, and evolved the above scenario as a way of providing minimal player input to control the state of these. The narrator gets more urgent the closer she gets to the moment she must leave, as the time of day advances with each scene; and choosing a location for each meeting decouples the scenes from a fixed environment. The satellite sentences would be in charge of contextualizing scenes on both these axes, reminding the reader (unobtrusively, hopefully) where a scene is taking place and when in the narrator’s final day it’s happening. So, for instance, the system might insert “The waves lap against the old pilings, insistent” between two sentences, or “For a minute silence fills the cool night air.”

The process of crafting a complete story with this system led to a number of

⁸A full write-up of this project can be found in Reed (2012b); the game itself can be played at <http://aaronareed.net/almost-goodbye/>

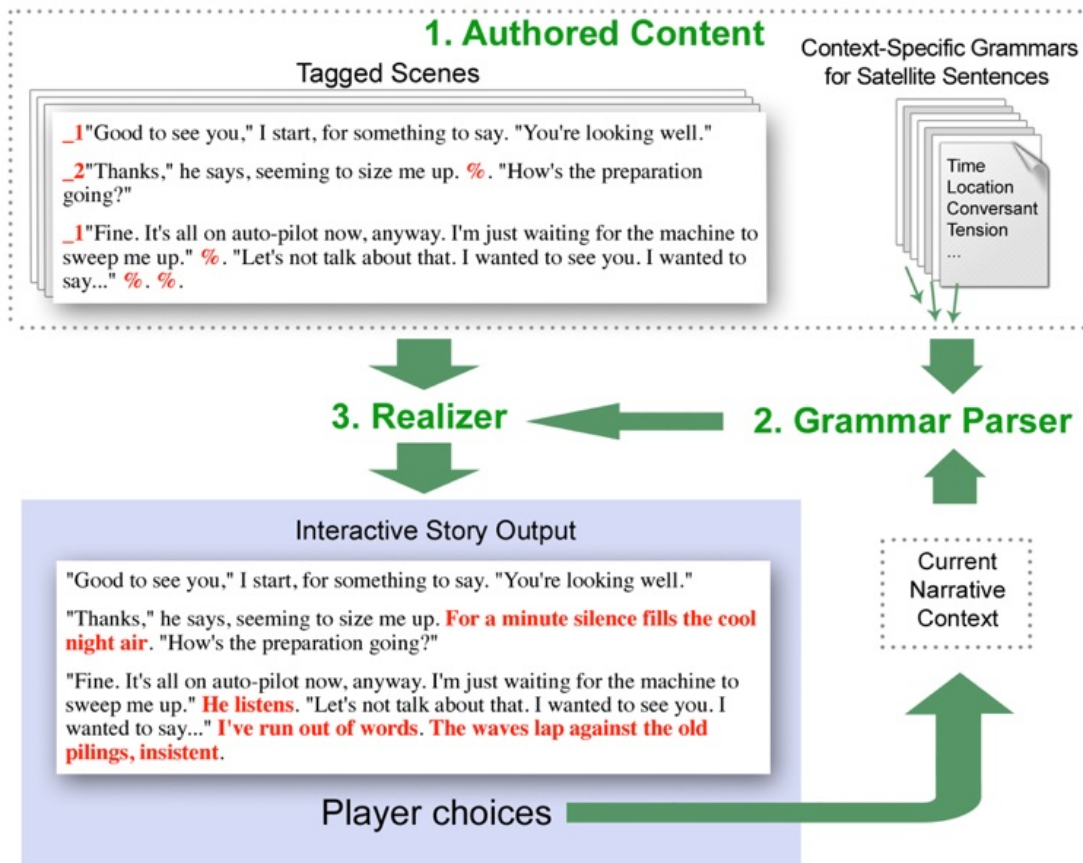


Figure 8.3: Basic system architecture for *Almost Goodbye*, from Reed (2012b). Player choices, not discussed here, vary the parameters the expansion grammars can use when building satellite sentences.

interesting insights. Originally, I expected to want more control over the specifications of each request for a generated satellite sentence, envisioning the need for tag markers to request sentences of a certain length, subject matter, or style. In practice, a simple unqualified tag tended to produce satisfactory results in most cases: in a sample traversal of the story, even with the extremely simple selection criteria described above, I tended to be authorially satisfied with the generation about 80% of the time. This is perhaps because the generator already takes into account some of the context that an author

would consider (and may in part be due to a reader’s tendency to pay less attention to satellite sentences than kernel text).

The ideal size of the component grammars was much smaller than originally anticipated. Most generative grammars have a large enough ruleset to allow for millions of possible expansions. Here, each individual grammar need only offer a few dozen to a few hundred possible expansions, leading to runtime compound grammars capable of making around five to ten thousand unique sentences. Since each individual grammar in the compound set is run only once or twice during an individual story segment, a smaller number of carefully considered variations proved more effective than a larger space of less curated alternatives.

Goodbye successfully demonstrated that minimal text generation could be unobtrusively integrated into hand-authored text—that a system could indeed be an effective collaborator, when given the right job to do.

8.6 *The Prisoner In Block Nineteen*

Billed cheekily as “a short experiment in natural language parsing and dynamic character simulation,” this game plays with a player’s ability to storywright even when the ideas generated for them do not necessarily have any coherence at all.⁹ At certain moments in a hand-authored story, the player can type out a yes or no question they’re putting to a prisoner in an adjacent cell. The narrative is structured to encourage the player to find deep meaning in the answers and attempt to connect them into a coherent

⁹This one can be found at <http://aaronareed.net/block19/> .

narrative. The answers, however, are completely random.

The project was inspired by a 1960s experiment by Harold Garfinkel¹⁰ where subjects were told they were asking yes or no questions to therapists via computer: even though the answers they got back were purely random, most subjects were able to construct stories explaining how even contradictory answers could be attributed to a thinking person on the other end. While framed as an example of the dangers of letting users think software is more intelligent than it really is, it also struck me as an example of the extreme lengths to which the human mind will go to storywright meaning into a set of lexia, no matter how arbitrarily assembled. If this set of things have been placed in the Wunderkammer, there *must* be a reason!

As an experiment, *Nineteen* is not always effective (in part because the narrative structure of the story requires writing *something* in response to an answer, which often falls down in practice) but is interesting to contrast with *Almost Goodbye*: minimalist contributions, from system and from humans, to an ongoing story.

8.7 *Archives of the Sky*

Archives is a complete analog storygame inspired by the games discussed in the previous chapter (especially *Microscope* and *Polaris*). It emerged out of my observation that most GM-less tabletop storygames abandon the idea of either a situated story (as does *Microscope*) or each player controlling their own character at all times (as does *Polaris*, where generally only one player character is in each scene). This is perhaps in

¹⁰Retold by Suchman (1987, p. 87) which made its way to me via Wardrip-Fruin (2009, p. 34).

part because the challenges of playing a character and enacting the other parts of a world (not to mention coming up with a coherent story) are difficult to take on simultaneously. With *Archives*, I wanted to see if there was a way to make this work. I also wanted to see if I could make a game centering around human conflicts between values, while keeping an epic scope.

Inspired by the epic science fiction of Alastair Reynolds, *Archives* begins with players designing a House, an eons-old order in a galaxy that has long since been filled by human colonization. A House will have certain fundamental values that cannot be compromised, developed collaboratively by players. Each player then makes a character who is a member of the House, with their own personal values. The mechanics of the game are designed to collaboratively evolve a story in which these values come in conflict with each other.

Play moves through a series of phases, with players sequentially making scenes in each one. In early phases, players can freely invent scenarios where their characters encounter old adversaries or unusual mysteries. A mechanic encourages players to connect these plot events to values. Eventually, a character may propose a *dilemma* which explicitly pits two values against each other. Players have various actions by which they can attempt to avoid the dilemma, or pick a side to fight for. The mechanics work such that players can make it more likely that they will be successful in the resolution of the dilemma, but can never be entirely sure. When the dilemma is resolved, characters who were forced to compromise a value must *adapt*, permanently changing some aspect of their character to reflect the emotional toll this compromise has on them. As in

Polaris, the story can continue until a player can no longer legally adapt, in which case their character's story is over.

Archives was useful for shaping my own understanding of how tabletop storygames can use mechanics to encourage interesting, coherent plots, and how the different creative impulses of everyone at the table can be drawn together and shaped without the authority of a gamemaster to arbitrate. The game is still in development, but an early version of the rules can be found in Appendix B.

Chapter 9

Conclusion

1. Do you wish to hear the story of the three alert peas?
if yes, go to 4
if no, go to 2

From “Un Conte à votre façon” (A Story as You Like It), Raymond Queneau, 1967

The yearning for interactive stories is very old.

Text that invites the reader’s participation dates back at least to the *I Ching* in China (Montfort, 2003, p. 66) used to procedurally generate fortunes and advice; but perhaps the first *fiction* to directly invoke the audience can be found in patterns of geometrically arranged Chinese poetry in the 2nd century A.D. At each node of the pattern, the reader may continue reading the poems along several different lines: one famous example of this style can reportedly be read over 40,000 different ways. Even older grids of Egyptian hieroglyphs have been found arranged in non-linear patterns, able to be read backwards or forwards (Higgins, 1987, p. 152, 169).

In the 17th century these works became more complex. In Europe, intricately

drawn labyrinth poems appeared, their maze-like patterns made up of words, with alternate forks and a multiplicity of possible readings; other poems were inscribed in elaborate lover's knot designs with no beginning or ending (Higgins, 1987, p. 11, 92). Juan Caramuel de Lobkowitz, a Cistercian monk, crafted "circular poems, in which the circles overlap to create multiple potential readings"; he also planned (but never built) various devices for nontraditional texts, such as "a cylinder for printing out endless poems" (42). Higgins describes these marvels as "poems to be solved; one looks at what typically is an arbitrary and unconventional array of letters, and discovers its process" (11). The word "process" is key, for while pattern poetry does not directly speak to the reader, what are its nodes but implicit if/else statements, and what is the reader doing but executing a process to read them?

These traditions and projects were beautiful, thoughtful, artistic instantiations of an idea that was easy to parody: that a process or an ordinary person could ever replace a poetic soul. In *Gulliver's Travels*, Swift's hero views at the grand academy of Lagado a mechanical engine that generates new ideas by mixing words together, such that "the most ignorant person, at a reasonable charge, and with a little bodily labour, might write books in philosophy, poetry, politics, laws, mathematics, and theology, without the least assistance from genius or study" (Part III chap. V). First performed in 1607, Francis Beaumont's play *The Knight of the Burning Pestle* opens with a scripted sequence where a supposed heckler from the crowd (called "the Citizen" in the script) interrupts the Prologue and clambers on stage with demands to hear a story about someone more like him: "I will have a Grocer, and he shall do admirable things." The

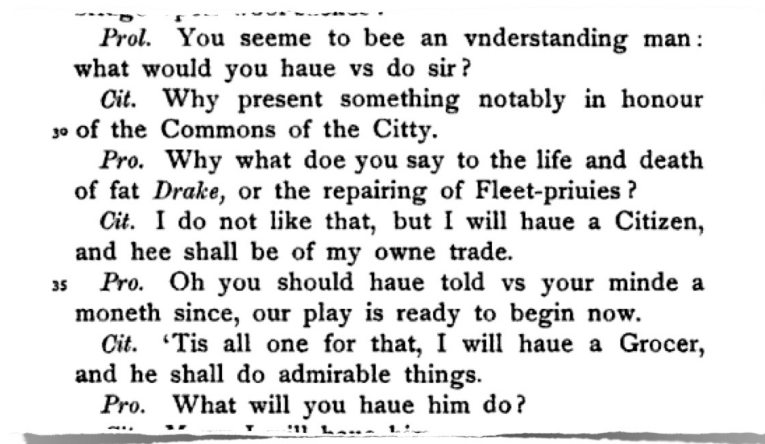


Figure 9.1: From *The Knight of the Burning Pestle*.

Prologue demurs: “You should have told us your mind a month since, our play is ready to begin now!” Before this dilemma can be satisfactorily resolved, however, the citizen’s wife also pipes up from the audience, eventually climbing on stage with her husband to help bargain for a plot more to their liking. Hundreds of years later, this comedic notion—that a grocer and his wife could tell a story as well as a playwright—drove Augusto Boal’s practice of a Theatre of the Oppressed, in which bringing citizens on stage and inviting them to build their own stories was both profoundly personally moving and a tool for social and political change.

Interactive stories can be effective without technology, but technology enables new dreams for how they might be told. “We regret having no access to machines,” said Raymond Queneau in the early 1960s of the Oulipo, which had been experimenting with interactive text for years; “this is a common *lamento* during our meetings” (qtd. in Motte, 1998, p. 51). Now we all have machines that can run such stories, and a growing body of design wisdom and code techniques with which to program them. But why are

these dreams worth pursuing?

One crucial reason is that interactive stories inherently involve processes of self-reflection and of creation. Even choosing between two options requires thinking in a fundamentally different way about a work’s traversal, and leaves a personal history behind. Interactors synthesize information, make decisions, and interrogate systems: the technological systems of the story, the ideas and options within it, and perhaps of who’s giving you those options and why. More deeply interactive stories can help us better understand how stories of all kinds are constructed, manipulated, and transformed. In an era of “fake news” and science denialism this becomes even more vital.

In this dissertation I have developed a critical lens for understanding how storygames work, based on the active development and release of a wide variety of playable experiments, and a detailed analysis of the technical and design infrastructure behind both effective and ineffective storygames. I will briefly summarize my major conclusions and contributions.

In chapter 1 I devised a framework for the analysis of storygames. I repurposed the term *expressive* to describe input that allows the player to communicate distinctive authorial intent. I also introduce the term *narrative logics* to describe the aspects of a playable system that control how a storygame’s lexia can be presented to and manipulated by the player. I used these analytical tools to examine the common “choice graph” structure in storygames, observing how the same narrative logic might work on multiple levels within the same game, and how minor technical changes to a logic’s implementation can have profound aesthetic impacts on the player’s perception of their

interaction with it.

In chapter 2 I studied the adventure game, one of the most famous (and famously failed) storygame genres. I built a practical bottom-up definition for this form based on an analysis of historical and contemporary definitions, revealing that the design space of adventure games includes kinds of work rarely studied together (such as text-based interactive fictions and point-and-click adventure games). I discussed the design pattern of the “eureka story” and how many classic adventure games struggled to achieve it because of the conflicting goals of emphasizing puzzles, exploration, and linear story. Finally, through a close reading of three contemporary games that descend from the adventure tradition I demonstrated how deep study of a seemingly anachronistic genre can reveal useful design insights about contemporary practice.

I next turned to a study of three emerging modes of storygame that move beyond the kinds of structures discussed previously. The first of these, discussed in chapter 3, is sculptural fiction, a mode that centers the construction (rather than navigation) of a graph of story nodes. Sculptural fiction encourages a more expressive, player-driven mode of engagement with a story space, and my design contribution is defining the space of this genre in depth, including how to build its systems, author content, and verify its output. My related technical contribution was discussed in chapter 4, where I provided a detailed case study of the sculptural fiction game *The Ice-Bound Concordance*, including details of the system’s novel engine for coherently combining story fragments.

The second emerging storygame mode I discussed was social simulation (chapter 5), in which decision-making agents consider an authored library of social norms to

determine what actions to take, enabling an emergent narrative outcome. I consider three major storygames driven by social simulations and determine what features are common to all three. I offer a detailed look at the considerations of authoring content for social simulations, which is very different from other kinds of interactive story authoring. I also discuss three major projects I contributed to involving social simulation: the game *Prom Week*, the Ensemble Engine that evolved out of it (in chapter 6), and the prototype Project Yarn that experimented with integrating social simulation into the design of an existing game (in Appendix A).

The final storygame mode I considered was collaborative storygames (chapter 7), games in which some of the lexia are generated by participants during play. I used Lars Konzack's Wunderkammer-Gesamtkunstwerk Model to study these games as centering around the computationally difficult activity of linking disparate story ideas into a unifying whole, and looked at ways systems might assist humans in doing this (rather than attempting to solve this problem directly, as other fields of interactive story research do). Finally, I consider existing digital experiments in this space and how these might continue to explore new forms of collaboration in future designs. I describe and list lessons learned from a number of my own such experiments (digital and analog) in chapter 8.

In conclusion, exploring and expanding the capabilities of interactive storytelling is important not only for the future of games, but to the culture at large. That exploration may lead across disciplines, to awkward spaces between the academy and industry, back and forth across the analog/digital divide and into the emerging spaces like augmented

reality that seek to erase it. The dozen or so released and playable projects of mine discussed in this dissertation each hope to advance the conversation around interactive story, drive new research into both design and technology, and inspire more storytellers to do the same. I'm excited to continue working to change the ways we think about playable media as a vital, necessary form of expression, and empower more players and creators to harness its potential.

Appendix A

Project Yarn

The social simulation engine developed for *Prom Week*, *Comme il Faut* (CiF), was later used in a second project I was involved with. An experimental collaboration between the UC Santa Cruz Center for Games and Playable Media, publisher Microsoft Studios, and game development studio Undead Labs, Project Yarn’s goal was to prototype how experimental story technology might be integrated into a commercial game. Specifically, we aimed to demonstrate how adding dynamic social simulation to an existing game could increase the player’s engagement with other characters and create a stronger sense of community. The project was organized through the Microsoft Studios University Program, and ran from November 2013 to June of 2014. I worked as one of the lead designers and coders on the project: other graduate students on the team were Paul Maddaloni and Ben Samuel, with academic advisor and Michael Mateas and industry advisor Richard Rouse III.¹

¹This appendix is largely the text of a previously unpublished whitepaper written collaboratively by the author and Paul Maddaloni, originally in fall of 2015; it has been lightly edited for inclusion here.

After some brainstorming and design sessions with Microsoft and Undead, the Santa Cruz team built a 2D prototype recreating abstracted core gameplay from the studio's 2013 game *State of Decay*, and combined this with a social simulation based on technology originally developed for Prom Week. While the original *State of Decay* let players manage a group of survivors during a zombie apocalypse, player engagement came more through traditional game mechanics than from interacting with the simple social state. Our prototype demonstrated a technical and design solution to making such a social situation playable and compelling enough to become a major part of gameplay. Through adding a social simulation that allowed for emergent character interactions, affected mission actions and outcomes, and offered players new actions and strategies related to monitoring and adjusting the social dynamics between game characters, we demonstrated that this approach can effectively be used to enhance a mainstream game.

Original Game

The original *State of Decay* features a group of survivors who establish a makeshift base during a zombie apocalypse. While characters can die and become more or less powerful during gameplay, the core game guides players through specific scenarios and a scripted plot arc that runs in parallel to free-form open-world survival gameplay. An alternate mode (“Breakdown”) lets players experience the game with a random cast, which omits nearly all authored story but allows for more flexible gameplay on each play-through.

While playing, the player controls one member of the base at a time and can switch between available characters between missions. In a typical gameplay loop, the player will take control of an idle survivor and prepare to go on a mission, inviting other characters along and collecting supplies from the base's stores. Fulfilling a mission involves navigating an open world, either towards a specific objective or with a more general goal in mind (such as opportunistically collecting resources or leveling up character skills). The world is populated with zombies, buildings that can be raided, additional bands of survivors, and other challenges or opportunities (such as climbing high towers to scout out the land). Upon returning to base, characters may need medical attention or rest, and gathered supplies can be used to better outfit characters or upgrade the base.

The original game includes some social mechanics, such as the possibility that a character with low morale will become unstable, requiring the player character take them on a short combat mission to ameliorate their emotional state. However, these sequences have only limited dynamism, and are not primary mechanics in the overall game.

Description of Yarn Prototype

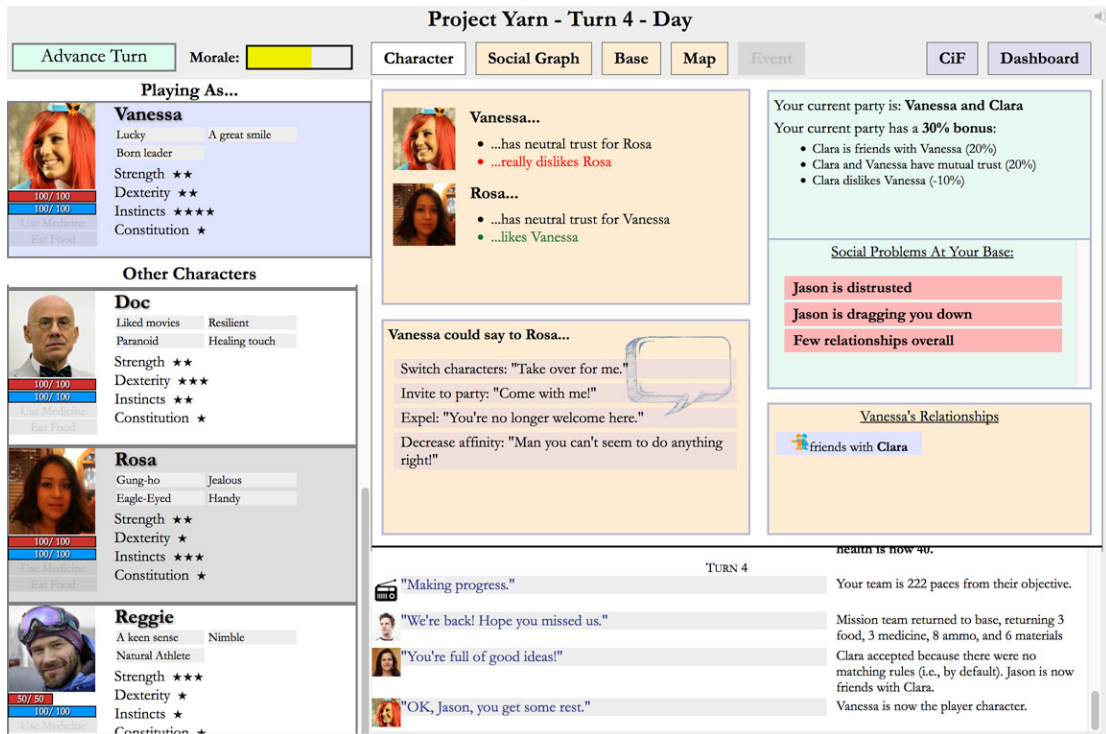
In the prototype, the player uses a 2D interface (Figure A.1) to simulate a turn-based recreation of the core *State of Decay* gameplay: assembling a team of survivors to go on missions to kill zombies, gather resources, and gain experience, then returning home and spending resources on maintenance and upgrades. New mechanics in our

prototype let the player see and influence the social situation at the base, such as having characters take social actions like becoming friends, and adjust the available gameplay options based on this state, such as having a character who distrusts a leader refuse to join a mission.

For Project Yarn, CiF was ported from Flash/ActionScript to Javascript, improving and simplifying it in several significant respects. (A full description of this project, eventually renamed Ensemble, can be found in Samuel et al. (2015b).) The prototype game was also written in Javascript, using the React framework to handle UI interactions and HTML/CSS for the UI itself. RequireJS for module handling and jQuery/jQueryUI were the only other major libraries used. The whole project was around 11,000 lines of code, with around 2,500 taken up by the CiF port, and an additional set of rules, missions, character definitions, and other authored textual content.

The prototype’s core game loop abstracts the resource gathering and management portion of *State of Decay*. The main goal is survival, which requires recruiting survivors and improving their skills, upgrading the base with better defenses, and defending against zombie attacks. Resources are consumed daily, so players will run out if they don’t head into the field to find more.

To keep the base harmonious and assemble an effective party, players need to keep their people happy. The social mood of the characters affects the base’s performance when dealing with random events, as well as characters’ effectiveness at performing tasks together (since solo missions are generally not effective). The prototype does not define an end-game state, functioning more as a sandbox than the original *State of Decay*’s



Turns: Beginning yarn-main.js main function.

Turns: Advanced to turn 1.

Turns: Advanced to turn 2.

Vanessa **rejected** the intent 'Vanessa is comradesVanessa wants to start being comrades with Jason'. Matching volition rules:

--> Someone who doesn't trust you is less inclined to become friends or mission partners with you. (weight -5)

Aggregate weight: **-5**

Clara **accepted** the intent '!...'. No matching volition rules, so took default action.

Clara **accepted** the intent 'Clara wants to increase trust Jason'. Matching volition rules:

--> People who aren't trusted are eager to gain trust. (weight +5)

--> Trust is hard to come by. (weight -1)

Aggregate weight: **+4**

Log Controls

☒ turns

☒ rules

☒ trigger

NONE ALL

Random Events (toggle)

Mission Events (toggle)

Switches

Toggle Skip Event Button OFF

Add Random Social State

Undo Last Turn

Restart w/Random Cast

Restart w/Scenario 1

Restart w/Scenario 2

Restart w/Scenario 3

Figure A.1: Screenshots from Project Yarn, showing the Character view in the main interface (top) and the Dashboard after a few social actions (bottom).

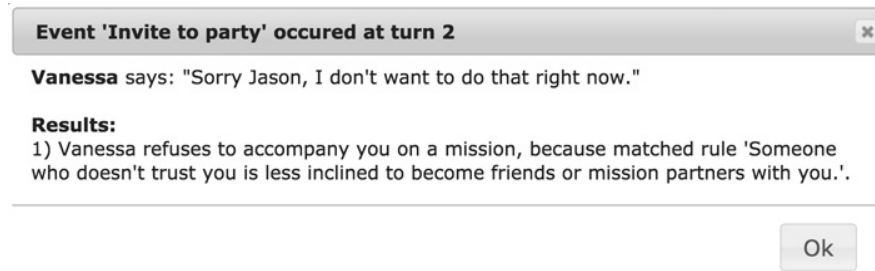


Figure A.2: Results of an action, showing social state causing gameplay consequences.

campaign mode.

The new social features change gameplay in several important ways. First, selecting any two characters lets the player initiate social actions relevant to the current feelings of those characters for each other. The allowable actions are calculated by CiF, which reconsiders character volitions each turn. While some of the prototype's set of actions act on the social state (such as increasing affinity with a friend by reminiscing about old times), with only indirect gameplay consequences, others have direct gameplay outcomes (such as kicking a disliked person out of the base).

The success or failure of these actions is again determined by CiF and based on the current social state. If the target character has, for instance, the directed status "hates" towards the selected character, they will be less likely to accept a request to become friends with the selected character. The prototype exposes the rules influencing both a character's desire to take a particular action and the recipient's decision to accept or reject its intent, by showing natural-language renditions of the most significant rules involved.

The social state also influences play during missions. Characters have numerical

modifiers to game skills based on the compatibility of their mission partners: for instance, inviting a romantic couple on a mission gives the party an aggregate bonus, while bringing out two characters who distrust each other leads to a penalty. Leaving people behind might have game effects, too, such as a jealous character starting to dislike the mission companions of their romantic partner. In another example of social state affecting gameplay, a mission leader who abandons their team to try to get back alive will find other base members having a significantly lower trust for them, which will affect their ability to make and maintain friendships, or find partners for future missions.

Another example of social state affecting gameplay was a system for generating missions based on social actions. For example, while playing as Sally, the player attempts to gain Rob’s trust. The prototype might generate a new mission characterized as having particular importance to Rob. If Sally leads a mission that successfully completes it, Rob’s trust for her will rise, though failure might damage this relationship even more.

The game interface exposes information about the social state in ways designed to help the player make gameplay decisions. A “social graph” (Figure A.3) gives a high-level overview of the entire social state, highlighting binary relationships (“friends”, “involved with”) and high or low values of numeric networks (affinity and trust). Character portraits show additional information, including both social and core-game statuses (“fatigued”, “angry at someone”) and permanent character traits that might affect the social state (two characters who both “like movies” have a better chance of getting along). A special window displays high-level problematic social patterns, such as a character being disliked by nearly everyone else, or the base as a whole having few

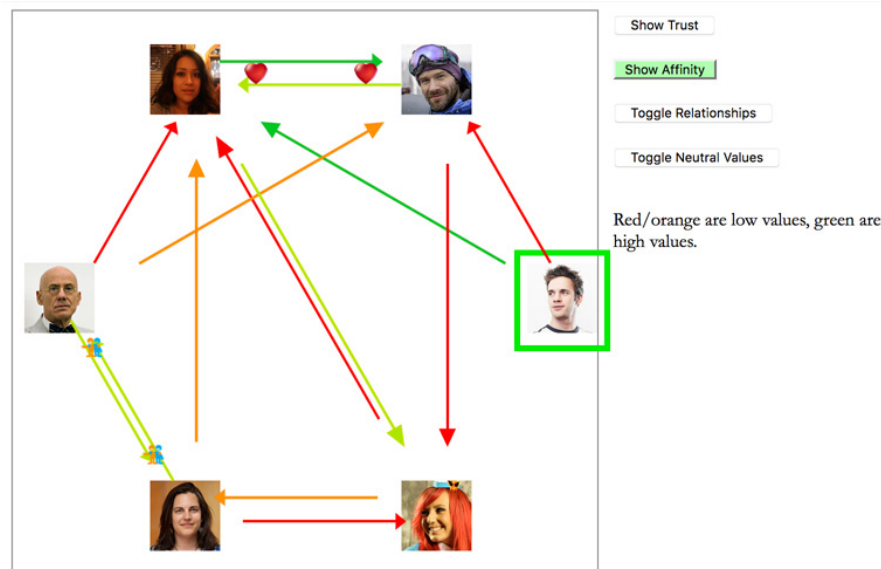


Figure A.3: A sample social graph in Project Yarn, showing affinity and relationships between the cast of characters.

beneficial relationships, and offers advice for how to improve the situation. Finally, a scrolling log window and detailed console view provide more details into how the social state changes turn by turn, and lets the player directly adjust the social state for testing or experimentation (if debugging is enabled).

The prototype provides several play modes to experiment with the social simulation, including a small set of predefined social scenarios that challenges players to “fix” a problem situation, and a sandbox mode where the game starts with a cast of random characters in a random social state.

While the prototype has not been formally tested, informal evaluation by both Santa Cruz and Microsoft/Undead teams was promising. Emergent play dynamics were discovered, such as sending a set of new characters out on “bonding” missions to build their trust with each other. In general, both teams felt that the two interwoven systems

provided a more dynamic home base game than previously, and better approximated the interesting social interactions one would expect given the game’s charged premise and potential for interpersonal drama.

Integration

The two systems (the minimal *State of Decay* recreation and the social simulator CiF) were designed to function independently of each other, communicating via a basic bridge module. CiF exposed a set of APIs that the prototype could call to recalculate the social state, request a list of appropriate actions for one character to take towards another, or update the social state based on game actions. This bridge module also performed CiF operations specific to the *State of Decay* prototype, such as adjusting the social state appropriately when a character died (removing active relationships, etc.).

By using this intermediary, both systems could run in parallel without having to communicate directly or be aware of each other’s existence. The reimplementations of CiF created for this project (now called Ensemble) is designed to be much more agnostic about the game it’s connected to, with a cleaner API and more flexibility to allow for different kinds of social simulation.

Drama Manager Versus Social Simulation

The initial project proposal discussed the possibility of including a drama manager on top of the systems described above as a stretch goal. While this did not

prove achievable in the limited time available for the project, we believe in a project with larger scope, drama management techniques would further enhance the emergent pleasures afforded by a social simulation. For example, a drama manager coupled to a social simulation might look for opportunities to introduce characters with highly conflicting traits at moments when the challenge seems too light (adding a “troublemaker” to the group), or could selectively adjust the likelihood of community-strengthening actions being accepted or rejected, tuning the amount of social unease up and down to fit a designer-mandated curve of rising or falling tension. We hope to be able to further investigate this combination in future projects.

Challenges

One of the most difficult things to accomplish in this project was to convey relevant social information to the player. Preliminary play-testing suggested players found it difficult to determine how their actions were influencing the relationships between characters. As in similar design iterations that took place during *Prom Week*’s development, we needed to find a way to give players the information they needed about a complex underlying social state without overwhelming them.

One step towards addressing this was to make pertinent information easily visible in the game’s Characters view. A brief synopsis of the relationship between the current lead character and the selected character summarizes the high-level state: two brief sentences are used per character to describe how much that person trusts and

likes the other. Hovering over that sentence shows a pop-up window with numerical details for each of these relationships (a value between 0 and 100, with higher numbers representing a stronger feeling).

Players can also quickly see basic social information for the selected character, with another window summarizing all their important friendships and romantic involvements. This lets players quickly see how popular a character is, and who might make a good match for a mission party.

The Characters view also has a section called “Problems at Your Base.” This uses some basic pattern-matching on the social state to highlight overarching issues across the whole base that might need dealing with, such as a character who’s universally despised. Hovering over each mentioned problem gives a more detailed description of what’s causing the problem, and a suggestion of possible ways to try resolving it.

In an effort to further players’ understanding of the current social state, another view called the Social Graph was constructed to quickly show all of the characters in the base, as well as their feelings toward, and relationships with each other. Player can look at the current trust or affinity between characters, shown with colored-coded arrows between characters. Numerical values between characters are also shown when hovering over an arrow. Other types of directed social parameters can be toggled on and off.

Lastly, in the Dashboard view, players can view a console which lets them interact directly with CiF or view the results of the most recent social calculation. As CiF reasons over the social state, it logs its reasoning to this console, and other information about the current state can be requested via a set of commands. While this view would

probably not be present in a released game, for the purposes of understanding the prototype it gives a direct pipe into CiF's behavior.

With all of these measures in place, we found that players had a much better handle on the social state of their base. Through this, players were more able to appreciate and enjoy the social interactions which made social engineering possible.

Project Successes

The team felt the final delivered prototype was a success on several major fronts. First, the CiF engine, originally designed in tandem with a specific game experience (Prom Week) was generalized and integrated with a game not originally intended to feature complex social simulation. Despite this, it still successfully enabled emergent stories connected to gameplay. Social interactions went beyond canned responses to become truly playable parts of the game experience. The social interaction not only provided its own layer of interest but became integrated with other existing game dynamics, opening up new design spaces for the prototype creators to explore.

Another success was the creation of a prototype that captured the core feel and dynamics of the original *State of Decay*. The demo game has base and resource management, missions that take place in the field, random events at the home base, and (the much improved) social interactions between characters. While it was an unknown factor whether a small student team would be able to create a recognizable facsimile of a complex AAA title, the resulting prototype hewed close enough to *State of Decay*'s

core game loop that it was easy to imagine how the integrated social simulation would improve the original game. This is a promising sign for future partnerships between games research groups and game studios.

Finally, the project was successfully completed in a relatively brief timeframe: about six months of part-time work from a small core team. In this time frame, the team re-implemented and improved the CiF engine in a new framework, recreated the core gameplay of *State of Decay* in a simplified standalone format, authored content (such as social rules and missions) to exercise the system, and refined the prototype to better serve as a standalone demo. The project was completed on time, and met all of the goals of the original proposal.

***Yarn* As Social Simulation**

As another example of a game driven by social simulation, it is illuminating to briefly compare *Yarn* to the pure social simulation games discussed in Chapter 5.

For instance, while in the previous chapter I made the observation pure social simulations can be read as interactive comedies of manners, with the player acting as a destabilizing agent violating social norms, this relationship is more complex in Project *Yarn*. First, the addition of a separate simulation (the open world with random events and depleting resources) that can interact with the social simulation already creates a greater source of churn and instability than a purely-social game. Combined with the design goal of increasing player challenges by deliberately upsetting the social state,

this tends to flip the player's role on its head: now the player is trying to smooth out social tensions and keep things running smoothly. The dramatic tension is now driven by external events. The drama is in how the player can overcoming increasing social tension to achieve their gameplay goals, not in how they create social tension as a source of interesting dramatic conflict.

Mixing social simulation with existing modes of gameplay produces a more complex and richer design space, the implications of which have only begun to be discussed: Yarn, while unfortunately unreleased, is one of the few attempts to deeply explore it. It is the team's hope that future games continue exploration of this unique hybrid space, and discover what play dynamics are enabled by adding social simulation to an existing game genre's toolkit.

Appendix B

Archives of the Sky

Archives of the Sky, discussed in (8.7), is a complete tabletop storytelling game developed during my research into collaborative storygames. This snapshot of the game's rules represents its state of development as of May 2017. Visit <http://archivesofthesky.textories.com/> for details on the release of the final game.

Setting

It was always our destiny to conquer the stars. It just took a bit longer than expected.

The speed of light proved an unbreakable limit. Interstellar voyages are a slow, lonely drifting between aching distant pockets of warmth and light. Our migration across the galaxy was crawling but inevitable, ripples on an unimaginably vast pond.

Most never took such a voyage, and most who did took only one. But a few strange souls chose a life of forever wandering, exploring the unmappable vastness one centuries-long voyage at a time. Spending most of their years in cryosleep, these wanderers lived on different time scales. Forever adrift from their homeworlds and outliving the civilizations they were born to, they gained a unique perspective. Combined with carefully guarded life-extension technology, they became nearly immortal. Eventually they formed their own social structure: the great Houses, humanity's most enduring institutions.

A million years have passed since then.

Now we have filled the galaxy. Humans have flourished, evolved, regressed, died out, and been reborn, thousands of times and in millions of variations. Vast empires spanning countless star systems have come and gone like the twinkling of sparks in a fire. But the Houses, suspended above it all like moths over that rippling pond, have endured. Each devotes itself to some core principle that goes beyond a single civilization or culture. Some seek knowledge or protect life. Others seek pleasures or thrills. And some record the history of mankind: the archives of the sky.

To play, you will need a stack of blank index cards, some pencils, a large clear tabletop, some books related to the setting (such as sci-fi novels), and three to five

players (including yourself). A sand or digital timer and some note paper are helpful on occasion. You can expect your first game, including building your galaxy, to take three to four hours. After that, continuing episodes with the same characters will take around two hours each.

Building

The galaxy is too large for any one player to design. Together, each player will continuously help imagine and remember the details that become important to your stories.

The setting establishes only one major ground rule about the universe: *nothing moves faster than light*. While you exist in an unimaginably vast cosmos, when facing danger you are also unimaginably far from help.

Your Trove

Your **Trove** is a deck of cards that you'll draw from for inspiration and resolution, filled with inspiring, evocative words. Create it now. Cut about five index cards per player in half across the shorter side and distribute these cards evenly. Give each player a book connected to the setting (epic science fiction) and set a timer for two minutes. While the timer runs, flip through your book and write down interesting words you see, one per card. Don't use simple words (a, the, an) or proper nouns (specific people or places). Try for a mix of verbs, nouns, and adjectives/adverbs, both positive and negative, all interesting or evocative.

When time is up, collect these cards together, shuffle them, and place them face down on the table. This is your Trove. At various times you'll be instructed to draw from it and do various things with the results. A player who needs inspiration can also, once per turn, draw from the Trove and use the card in any way they wish to spark their creativity.

The Trove is a face-down deck: you will draw the top card, but should not know in advance what it is. When you exhaust the Trove, reshuffle it.

Your House

Your characters are all members of one House, a group of near-immortal humans who skip like stones across the galaxy's history, one decades-long star voyage at a time. Before you create them, you must first create the House they belong to.

Your First Values. Decide collectively on your House's ultimate purpose, overarching mission, or reason to exist. Try to keep it short: a verb, or a verb and a noun, phrased as an action statement (i.e., "We hunt"). Write it in large letters on an index card (horizontally, as for all Values).

Your House's second Value is always this: **"We will always remain human."** As with all Values, arguments over how this should be interpreted will be a source of conflict. Write this Value too on a card and place it beside the first.

Your Place. Discuss briefly how your House is seen by others in the galaxy. Are you strong and influential? Quiet and in the shadows? Honorable, mysterious, feared, ignored? A word or two is enough. Write this on the card with your first value, underneath.

Three Others. Devise three Others of equal or greater power: an old **Ally**, a new **Opponent**, and a growing **Mystery**. These should be of greater scope than your run-of-the-mill galactic empires that rise and fall in a single turn of the galaxy. They should be forces your House has or will have to deal with across millions of years: other Houses, post-human intelligences, unknowable alien forces, meta-civilizations, forces of nature, cosmic mysteries. Everyone should take brief notes on these three Others: just the name and a few words about their primary qualities is fine.

Two More Values. Propose two additional Values for your House. Some of these may solidify ideas that have already come up in discussion; others may be new ideas. Everyone must agree on a Value for it to be accepted; discuss and revise if necessary. Write a card for each.

Values should begin **"We always..."** or **"We never..."** (or some equivalent: they should never be wishy-washy). They are ideals that everyone in your House has sworn to live by.

Picking the right values is important for a good story. Apply the following two questions to any Value:

- Is this Value a belief about what is right?
- In the right circumstances, would a member of your House die to defend this Value?

If the answer is an easy yes to both, it's a good Value. If you're not sure, refine it until it becomes an easy yes, or pick a new idea.

Decide why your group has come together. Maybe you have specialties within your House's purpose, maybe you're doing something only tangentially related to it. Maybe you're from different factions within the House and have things to teach each other. Maybe you're on a mission connected to one of your House's Others.

Finally, name your House. Choose an evocative word connected in some way to your values: for example, a house that never compromises might be the House of Stone. Write the name on a new card, vertically oriented, and place it and your House Values on an edge of the table (as if it were another player), keeping the center of the table clear.

Your Characters

Begin with your character's name. If you need inspiration, draw from the Trove and either use what you draw directly, or as indirect inspiration: rearrange the letters, or make something that sounds similar or is suggested by your draw. If you're still stuck, hand another player the card you drew and ask them to suggest a name for you.

Write your name in large letters on the upper half of an index card, then turn it over and write it again for yourself on the bottom half. Fold it in the middle so the card stands up.

Personal Values. You'll define your character by creating a few Personal Values. These are things your character has come to strongly believe, across millenia of traveling the galaxy with their House. In a three player game, each character should have three personal values; for a four or five player game, each character should have two.

Make a new card for each Value and keep them spread out in front of you: all values should be visible on the table at all times. Phrase a value starting with your character's name, and use **always** and **never** words (or some equally strong equivalent) just as with House Values. Apply the same two criteria and rewrite the Value until both answers are a strong yes:

- Is this Value a belief about what is right?
- In the right circumstances, would your character die to defend this Value?

Traits. Each Personal Value should also have a positive Trait on it which connects in some way to the value. The connection might be indirect, but should exist. A trait can either be about your personality, or an area of expertise or interest.

Some sample personality traits: *Trusting, Shrewd, Honest, Noble, Loyal, Kind, Careful, Energetic, Generous, Whimsical, Courteous, Intense, Innocent, Brilliant, Hilarious, Prepared.*

Some sample expertise traits: *Memory, Diplomacy, Deception, Music, Empathy, Conversation, Moon-surfing, Research, Stargazing, Strategy, Engines, Athletics, Tinkering, Weapons.*

Decide who will be the first Narrator, which can be anyone who has an interesting idea. The Narrator should write "Narrator" on a blank card and place it before them. The player to their right should do the same for the title "Epic," and the player to the left the same for the title "Intimate."

Your world is seeded. You're ready to play.

Playing

Each game of *Archives* will tell a single Episode in the lives of your characters, unfolding in a series of Scenes staged by each player in turn. Most episodes will begin

with your characters arriving in a new star system after a voyage of decades or centuries, and finding something unexpected or mysterious there.

An episode moves through four phases, which you can roughly think of as the three acts of a story and its denouement:

The **Exploration Phase** is about finding an intriguing situation, asking questions about it, and fleshing out its consequences.

In the **Dilemma Phase**, you'll face a problem arising from the situation which puts two Values in conflict, and come up with a plan for dealing with it.

The **Climax Phase** is an action-packed attempt to execute the plan without compromising the threatened Value.

Finally, in the **Resolution Phase**, characters who had to act against a Value may permanently change as a result, sometimes ending their story.

On your turn, you'll become the Narrator, driving a Scene that moves the plot forward and explores how it questions your values. Everyone, not just the narrator, can help tell the story of a scene, either in character or by providing colorful, intriguing details; but the Narrator decides how the scene begins, when it ends, and the primary contribution to the story that it's making. When a scene ends, the player to the left becomes the new Narrator, and so on until the Episode is over.

Scenes

In a Scene, everyone collaboratively tells the story, speaking as their character, narrating something happening, or injecting interesting or colorful details. One player, advancing clockwise around the table, is always the Narrator, taking the helm of the story with a specific idea for how to advance it. Other players may have other roles.

Setting the Stage. A new Narrator first determines what Action they would like to play from the appropriate list for the current Phase. Once they have one in mind, they set the stage for a scene, saying:

1. Who is in the scene
2. Where it takes place

The Narrator may play their own character in a scene, perhaps narrating them initiating a discussion or making a discovery; or they may describe external events for the other characters to react to, perhaps including contact with an NPC for the Narrator to play.

The *mechanical goal* of each scene is for the Narrator to reveal an Action. The *storytelling goal* is for everyone to play their characters, react to what happens, invent interesting details, and keep the story alive and coherent. All players may freely roleplay as their characters or invent details of the story world, but only the current Narrator should steer the plot and introduce major new developments.

If a Narrator is stuck for ideas, they may draw from the Trove and use it for inspiration, or show their draw to the group and ask for ideas.

Once the scene begins, the Narrator may reveal their chosen action immediately, or slowly build up to it, at their discretion. Once the action has been revealed and the other characters present have had a chance to respond to it, the Narrator should bring the scene to a close as soon as possible.

While all players are free to improvise during a scene, the Narrator has dramatic authority. If a player says something that contradicts a Narrator's intention for the scene, the Narrator may point to their Narrator card and say something starting with "But..." to move things back on course, even contradicting something another player just said: this is called **correcting** a scene.

The Epic and the Intimate

The players on either side have a specific job to do during the Narrator's scene.

To the right of the Narrator is the **Epic**. This person plays their character, if present, but their job is also to make sure the story stays grand in scope. This may take the form of additional clarifying details that extend the scope or grandeur of the story. ("The ship isn't just big: it's *miles* long.") The Epic may also feel free to make up details that increase the stakes of a plot development ("It's not just this one colony in danger. It's the five hundred million people across the planet!").

Don't overdo it, especially if the Narrator is already being sufficiently epic, but look for opportunities to inject a sense of wonder and awe at the scale of the character's adventures.

To the left of the Narrator is the **Intimate**. They also play their character, but their job is to make sure the story stays connected to the humanity of the characters and their values. The Intimate can narrate how their character feels about what's going on, or ask other players to do the same for their characters. They should also add small details of setting and character: the texture of the alien monolith, the sound of the phase weapons, a bead of sweat on an NPC's forehead, a small smile only a few people see.

As the Intimate, look for opportunities to keep the story grounded in human senses, emotions, and details.

All players in a scene should feel free to invent exciting details and use interesting specifics. Rather than stop the scene to ask what's allowed or expected, they should build and extend the world and flesh out its details at every opportunity, and trust the Narrator will correct the scene if they go too far.

Scene Order:

1. Pass Role cards left.
2. The new Narrator picks an Action appropriate for the Phase, and decides what their scene will be.
3. The Narrator sets the stage, setting the location and cast.

4. Scene begins. Narrator works towards revealing their action; Epic and Intimate play their roles; everyone with a character plays them and adds interesting detail.
5. Once the action has been revealed and its immediate consequences made clear, the Narrator should adjust cards and bring the scene to an end.
6. The scene is over. If the conditions for moving to the next phase are met, do so now.

Marking a Trait

At various points during the play, you'll have an option to *mark a trait*. This means making a check mark by one of your character's positive traits (connected to your personal values). If you wish and it makes sense, you may narrate how the trait connects to the scene. At the end of an Episode, all traits are unchecked.

You may not mark a trait you've already checked this episode, nor may you spend a negative trait (which you might have acquired after Adapting in a previous episode). If you have no unchecked positive traits, you may not mark a trait.

Exploration Phase

During this phase, the players introduce Questions that beg investigating, and explore how those questions connect to their values. Any player with a good idea can begin an episode by Asking a Question (which is initially the only action available).

<i>Action & When Allowed</i>	<i>Effect</i>
Ask a Question <i>when fewer than 3 Questions</i>	Create new Question card
Complicate a Question <i>at least one Question</i>	Rewrite a Question card
Threaten a Value <i>at least one Question</i>	Move a Value forward
Propose a Dilemma <i>At least two threatened values</i>	Create or modify a Dilemma

The Exploration Phase ends when someone creates a question that everyone agrees is a valid dilemma.

Action Descriptions

Ask a Question. When there are fewer than three Questions in play, the Narrator can stage a scene to *Ask a Question*. A question, especially an episode's first, should be an intriguing situation that requires exploration and investigation.

It's okay (and, indeed, preferable) to ask a question without knowing the answer.

When finished, the player should write the question on a vertically-oriented card as succinctly as possible and place it in the center of the table. Write large and legibly.

Further questions can either build on the initial question, or be seemingly unrelated events that other players might try to tie together.

Questions should always be simple and clear, without compound parts, and with room for further investigation.

If a Narrator is stuck for ideas, they can draw from the Trove, and make up a completely random question inspired by their draw. (Perhaps they draw "signal," and make up the question "What is the mysterious signal?") They can then stage a scene involving a few other characters and invite them to improv something based on the prompt (presumably discovering a mysterious signal, and possibly tying this into what's already been established.) If anyone is stuck for ideas for more than a few moments, other players should gently encourage them to try this technique.

Complicate a Question. When at least one Question has been asked, the Narrator may stage a scene that complicates it, either revealing new information that leads to a revised question, or answering it in a way that leads to a new question. When you Complicate, make a new Question card and place it on top of the one you're complicating.

Sometimes in the course of play, a Question may seem to be answered in the process of some move being played, without necessarily implying a new question. At the end of a scene, if everyone agrees, you may write an answer on a Question card. Leave it on the table; a new Question may be played on top of it later. You may not answer the last unanswered question this way.

Threaten a Value. When at least one Question is in play, the Narrator can explore how any Value on the table is affected by the current circumstances.

In a Threaten scene, the Narrator picks a value—their own, someone else's, or a House Value—and always asks the same question about it:

Is this Value threatened by what's happening? Why or why not?

As with any scene, the Narrator sets the stage to say who appears in the scene and where it takes place. However, a Threaten scene can have a maximum of two characters, at least one of whom must be a stakeholder in the Value under discussion.

It's the Narrator's job to keep the scene on track towards answering whether and how the Value is threatened, and ending the scene as soon as this has been established. If the scene is dragging on, it's probably because at least one character thinks the value

is threatened, and the Narrator should move to end the scene with that conclusion. For a personal Value, however, it's always the choice of the player who owns the Value whether it becomes threatened or not.

If a Threaten scene ends with a Value being threatened, it should be pushed forward towards the center of the table. This Value is now eligible for being part of a Dilemma.

Note: it can be hard to keep track of everybody's personal values. A Narrator stuck for ideas can ask if anyone has a value they feel might be threatened by the circumstances. Players should always be thinking about how their characters are reacting to the situation through the lens of their values.

If an additional player character wants to enter a Threaten scene, they can do so by marking a trait.

If the Narrator wants to ask or answer a question as part of a Threaten scene (introducing major new plot events), they can do so by marking a trait.

At the end of any scene, a player may attempt to pull back a Value that was threatened in a previous scene (before the one that just ended), if they no longer feel it to be threatened by the developing story. All players must consent to pull back a Value.

Propose a Dilemma. See the next section for details on dilemmas and this action.

Dilemmas

A Dilemma is a major decision between two mutually exclusive courses of action, a situation which your characters, as representatives of their House, must soon resolve. Only one Dilemma can be in play at a time. Each side of a dilemma must be supported by a threatened Value. This implies that choosing one course of action will threaten the Value of the other side.

The Dilemma sheet available online can help organize a dilemma. Whether you use the sheet or not, you should place the Dilemma Question in a key central position, and establish space on either side of it as corresponding to the two courses of action. You should place the Value connected to that course of action on the appropriate side. The action threatens the Value on the opposing side.

A Dilemma should not be created until a Narrator has a clear sense of how the plot is pushing the characters towards a conflict between two threatened values. If the Narrator is not yet clear on this, they should instead play another kind of move to get more clarity about what's at stake.

When a Narrator plays Propose a Dilemma, decide as a group which threatened value will be attached to each of the two potential decisions. While the dilemma might connect to any or all of the threatened values, and all of them might be part of the story, mechanically you can connect only one Value to each side of the Dilemma. Out of character, you should attempt to find the pair which most starkly defines the emotional stakes of the Dilemma: which best creates a situation where one Value cannot be honored without trampling on the other.

To advance to the Dilemma phase, the Dilemma must be *understood*, *irreconcilable*, and *conflicted*.

- **If not every player understands how the Dilemma represents a mutually exclusive conflict** between two courses of action (and a parallel conflict between two Values), the Dilemma is not *understood*.
- **If splitting the party or delegating one action to some NPCs would resolve the Dilemma**, it is not *irreconcilable*. Your characters are representatives of your House and must act unanimously in its name: you must pick one course or the other.
- **If the characters are already in unanimous agreement** over which course of action to take, the Dilemma is not *conflicted*.

If any of these are not true, resolve the issue as a group, or set the Dilemma aside for now and continue playing Exploration scenes until you can, possibly with a different Dilemma.

The Exploration Phase ends when a proper Dilemma has been created.

Dilemma Phase

During this phase, continue passing around the role of Narrator as before, but the set of actions has now changed. The goal of this phase is for the characters to choose a course of action for resolving the Dilemma.

When the Dilemma Phase starts, reclaim any of your threatened Personal Values on the table that are not part of the Dilemma.

<i>Action & When Allowed</i>	<i>Effect</i>
Vote	State or change your decision
Perspective <i>you have an unmarked positive trait</i>	Attempt to sway votes.
Overtake <i>you have an unmarked positive trait</i>	Propose a drastic plan to sidestep the Dilemma.
Force Our Hand <i>you have a threatened Resolved Value</i>	Immediately change all votes in your favor.

The Dilemma Phase ends as soon as all votes are on the same side, for any reason.

Action Descriptions

Vote. Simply announce your character's current or new vote, for one way of resolving the Dilemma or the other. Whenever a character's vote is stated or changes, move their nametag to the appropriate side of the Dilemma card. If it's not clear why your vote has changed, you may narrate a brief explanation; but long discussions or revealing new information should take place during a Perspective action.

It's initially okay to vote against a House Value or even your own Personal Value, but at the end of the episode your character might face lasting consequences for any Value they share that was compromised. After the first episode, you may have Resolved Values you are not allowed to vote against.

Perspective. If you are able to mark a trait, you may do so to narrate a scene with the goal of trying to change other characters' votes. You may do this either through verbal persuasion in character, or through revealing new information as the Narrator. You can't contradict anything previously established, and in particular you should not alter the Dilemma in a way that invalidates the threat to either Value, but you can up the stakes, or reveal a new moral complexity in a decision.

As always, the Narrator should drive this scene towards answering a question, in this case *Does the argument or new information sway any votes?* They should end the scene as soon as they feel it's been answered, or if things seem at an impasse. At the end of a Perspective scene, any player may choose to change their vote.

Overturn. If you are able to mark a trait, you may propose a plan that would allow the characters to avoid facing the Dilemma directly: a daring rescue to change the terms, a desperate move that achieves two seemingly contradictory goals. The Overturn will put all players at risk: if it fails, everyone will face the consequences of both Values being compromised. All player characters must be involved in the scene, and all must agree to the plan for the Overturn to progress. If anyone dissents at the scene's conclusion, the Overturn does not proceed. If it does, however, return all nametags to their players and immediately begin the Climax Phase, using the special Overturn instructions.

Each player may only attempt an Overturn once per Dilemma. An unsuccessful Overturn still costs the Narrator the trait they marked to attempt it.

Force Our Hand. A player with a Resolved Value at stake in this Dilemma may take this action to do something irrevocable that immediately changes all votes to the side of their choosing. This represents a drastic action, perhaps behind the other characters' backs, that commits to one side or the other, such as launching a missile, or hijacking a ship. This may have serious repercussions for your character's future in the House.

Note this action can't be chosen during the first Episode, since no players will have Resolved Values yet. When Force Our Hand is played, move all votes to the appropriate side and immediately begin the Climax Phase.

If every player has taken a turn since a vote last changed, and the vote is still split, the next Narrator *must* narrate a crisis which acts like an involuntary Force Our Hand: a situation has arisen that's caused one side to be chosen over the other. Time has run out; a malfunction cuts off one possibility; an NPC takes a drastic action taking the decision out of the players' hands. Pick a side that seems natural in the fiction, or choose one at random, and update votes accordingly.

As soon as all votes are on the same side for any reason, move on to the Climax Phase.

Climax Phase

The characters, voluntarily or not, are now committed to a course of action: defending one value while risking another (or, for Overturn, attempting to defend both and putting both at risk). Each player gets to be Narrator once during this phase, spinning an exciting scene where their character contributes to the mission or tries to answer the question of whether any values have been compromised.

<i>Action</i>	<i>Effect</i>
Advance	Try to complete a piece of the plan
Protect	Try to prevent the threatened Value from being compromised

Standard Climax. (For Overturn, see below.) First draw two cards from the Trove. Write SUCCESS on one and FAIL on the other. Turn them face down and randomize them. Put one under the Dilemma, representing the state of the *Plan* of action, and put the other under the *Value* threatened by that plan (the one on the opposite side from the course of action chosen).

On a Narrator's Climax turn, they'll decide whether they'd like to try to **Advance** the plan, attempting a dramatic action to move it closer towards successful completion, or **Protect** the value it threatens from being compromised. In both cases the procedure is the same:

Narrate your scene, bringing it to a moment of climax where you're about to either succeed at a critical task or fail. Then, the player to your right draws from the Trove, showing the card to the group. They decide whether to interpret the draw as a success or a failure. They narrate the result, write either SUCCESS or FAIL on the card, and add it to the appropriate deck: the stack under the Dilemma, if the Narrator was advancing the plan, or the stack under the Value, if the Narrator was trying to protect it.

If the result is a FAIL, at any point before the card is placed in the result deck a third player (neither the Narrator nor the person to their right) can mark a trait to change it to SUCCESS, narrating how their character helps reverse the seeming failure (ideally involving the trait somehow).

Once all players have narrated a Climax scene, if the next player in sequence is still able, they may choose to mark a trait to narrate one final climax scene of either type, resolving as described above.

The moment of truth has now been reached. The new Narrator shuffles the two success/fail decks, then flips over one card from each.

1. **If the Plan succeeds**, the players were able to execute it successfully. A **failure** means they were only partially able to execute, something unexpected interfered, or it failed entirely, depending on what's appropriate for the fiction, with disastrous results.
2. **If the Value succeeds**, it wasn't seriously threatened, or the worry it represents does not prove true. A **failure** means the Value is compromised or the worry comes true in the worst possible way.

The Narrator who drew the cards now stages a scene explaining the final result: why the plan succeeded or failed, and why the value held firm or was compromised. They may use the Trove words on the drawn cards for inspiration, or ignore them. The Epic and Intimate are free to chime in, as always.

An Overturn Climax works as above except with a single success/failure deck that everything rides on. Seed the single deck with two cards, one SUCCESS and one FAIL. Players may still choose between advancing the plan or protecting values, but the result card in both cases goes into the single deck. At the end of the phase, draw once for the result. A **success** means the Overturn worked: both goals were achieved and neither value was compromised. A **failure** means disaster: both goals failed or were compromised, as were both Values.

Resolution

Continue in turn order through one more rotation. Each player may now play a final scene (or simply make a short narration) about how things turned out, both in the setting, and for their own character personally. If you choose to or must Adapt, do so during your turn.

1. **If the Plan failed**, anyone may choose to Adapt.
2. **If the Value failed**, anyone sharing that Value must Adapt.
3. **If both failed**, everyone must Adapt.

Adapting. When you Adapt, your character has changed as a result of the events of the Episode, moving you either closer to or further from your House. Pick one of these options:

1. **Let Go.** Permanently discard one of your Values. Explain why your character no longer believes in it, and why your heart is no longer in its associated trait. You may not Let Go of a Resolved value.
2. **Resolve.** Pick a Value, and either write your character's name on it and underline it (for a House Value) or underline the "always" or "never" (for one of your Personal Values). This Value is now Resolved for you. You are even more committed to it: you can never vote against it or Let Go of it. Explain why your character is now resolved to never (or never again) compromise it.
3. **Twist Up.** Pick a positive trait from one of your Personal Values and cross it out, replacing it with a negative one. You may no longer mark this trait. Explain how your experience has twisted your worldview.

If you have Let Go of your last personal value, or if you cannot legally Adapt in any of the above ways, or if you so choose, your character's story is now over. Include in your Resolution scene's narration a description of how they've become even more committed to the House, or have lost their faith in it, and where they go from here.

Never Adapt more than once per Episode, even if multiple Values were compromised.

The only way you can avoid adapting is if you can mark *two* traits. Describe in your resolution scene how you find the strength to avoid letting the events of the Episode affect you—this time.

If someone played *Force Our Hand*, on their Resolve turn they should choose whether their character is now permanently cut off from the group (dead, departed, etc.) or would like to rejoin. If the latter, allow everyone to make a brief statement, then vote by secret ballot. On a unanimous approval, the character can rejoin. Otherwise, or if the player chose to voluntarily leave, the player should narrate an ending to their character's story.

Wrapping Up

1. Erase the checks from any marked traits.
2. Permanently discard the Dilemma's success/fail Trove cards.
3. Permanently discard the Dilemma card, and return Value cards to their owners.
4. If you think you might play again with these same characters, follow the instructions in Playing Again below.

Playing Again

If you intend to play another Episode later with the same characters, you should create an Archive deck. Transfer your three Others to three index cards, and also make one for any other significant forces introduced in your first Episode. At the beginning of each new Episode, review the cards in your Archive as a group. At the end of each Episode, create cards for anything new and interesting, and discard any cards that seem resolved or no longer of interest to anyone. Store the Archive along with your Value cards and character nametags to return to your galaxy later. (You should create a new Trove each session, although you might want to keep a few favorites to seed it for your next game.)

Archives of the Sky will tell a story about how the new forces in the galaxy either bring your characters closer to their House, or tear them away from it. Inevitably, one or the other will happen to each character, ending their story. A player can make a new character when this happens, or play without a character (see below). The group might decide to end the overall story as soon as any character's arc concludes, or when multiple characters wrap up in a single episode.

Playing Without a Character

If your character's story ends, or if you have four or five players, someone may choose to play without a character. They may play as normal, focusing on the plot and how it impacts the player characters:

- During the Exploration phase, stage question scenes that advance the plot or where interesting NPCs interact with the characters, or stage scenes challenging characters with potentially threatened Values.
- During the Dilemma Phase, stage scenes that clarify the stakes of the dilemma or invite opposing characters to hash out their differences.
- During the Climax Phase, narrate a complication to the plan, or a threat to the dilemma, and allow anyone playing a character to step up to address it. (You may act as the drawer and judgment-passer of the Trove card for that character.)
- During the Resolution Phase, narrate how things turn out for interesting NPCs or the situation as a whole.

Archives Acknowledgments

Thanks to playtesters Matthew Balousek, Duncan Bowsman, Jacob Garbe, Johnathan Pagnutti, Ben Spalding, Phoenix Toews, Tiffany Wong, and Wilson Zorn.

Thanks to the many storytelling games that have inspired me, including *Polaris*, *Fiasco*, *Microscope*, *Shock*, *Lovecraftesque*, *Kingdom*, *Dungeon World* and *Apocalypse*

World, Hillfolk, Nobles of Amber, Downfall, The Shab-al-Hiri Roach, Fate, Dream Askew, Dogs in the Vineyard, and Juggernaut.

And finally, thanks to Alastair Reynolds for his mind-blowing book *House of Suns*, which inspired this game about remaining human against the epic scope of the cosmos. Also highly recommended are his *Pushing Ice*, *Blue Remembered Earth*, and the short story collection *Galactic North*.

Ludography

Below are listed all the games and playable artifacts mentioned in this dissertation. The credited designer(s) are listed for each, followed by the studio and publisher (if any). The platform(s) the game initially launched on are listed next, along with launch year; platforms may be hardware, or a virtual machine with implementations on multiple platforms.

18 Cadence. Aaron A. Reed. HTML/Javascript. 2013.

80 Days. Joseph Humfrey, Jon Ingold, and Meg Jayanth. Inkle. iOS. 2014.

Adventure. Will Crowther and Don Woods. FORTRAN/PDP-10. 1977.

Alabaster. Emily Short (with John Cater, Rob Dubbin, Eric Eve, Elizabeth Heller, Jayzee, Kazuki Mishima, Sarah Morayati, Mark Musante, Adam Thornton, and Ziv Wities). Inform 7/Glulx. 2009.

Almost Goodbye. Aaron A. Reed. HTML/Undum. 2012.

A Mind Forever Voyaging. Steve Meretzky. Infocom. ZIL. 1985.

Angry Birds. Jaakko Iisalo. Rovio Entertainment/Chillingo. Mobile (iOS/Maemo). 2009.

Apocalypse World. D. Vincent Baker. Lumpley Games. Printed rulebook. 2010.

Archives of the Sky. Aaron A. Reed. Game rules (in progress). 2017.

- Bad News*. James Ryan, Ben Samuel, and Adam Summerville. Installation driven by “Talk of the Town” social simulation. 2016.
- Balance of Power: 21st Century*. Chris Crawford. StoryTron/Java. 2009.
- Betrayal at House on the Hill*. Rob Daviau, Bruce Glassco, Bill McQuillan, Mike Selinker, and Teeuwynn Woodruff. Avalon Hill. Board Game. 2004.
- Black Closet*. Georgina Bensley. Hanako Games. Ren’Py. 2015.
- Blackwell* (series). Dave Gilbert. Wadjet Eye Games. Windows/OS X/iOS/Android. *The Blackwell Legacy* (2006); *Blackwell Unbound* (2007); *Blackwell Convergence* (2009); *Blackwell Deception* (2011); *Blackwell Epiphany* (2014).
- Blue Lacuna*. Aaron A. Reed. Inform7/Glulx. 2009.
- Broken Age*. Tim Schafer. Double Fine Productions. Windows/OS X/Other. 2014 (Act One); 2015 (Act Two).
- Capes*. Tony Lower-Basch. Muse of Fire Games. Printed rulebook. 2005.
- The Cave of Time: Choose Your Own Adventure #1*. Edward Packard. Bantam Books. Gamebook. 1979.
- Coffee: A Misunderstanding*. Dietrich Squinkifer. Installation. 2014.
- Counterfeit Monkey*. Emily Short. Inform 7/Glulx. 2012.
- Crusader Kings II*. Henrik Fåhræus and Christopher King. Paradox Development Studio/Paradox Interactive. Windows. 2012.
- Dear Esther*. Dan Pinchbeck. Thechineseroom Limited. Source mod. 2008.
- Downfall*. Caroline Hobbs. Less Than Three Games. Printed rulebook. 2015.
- Dragon Age: Origins*. Brent Knowles, Mike Laidlaw, and James Ohlen. BioWare/Electronic Arts. Windows/PlayStation 3/Xbox 360. 2009.
- Dungeons & Dragons*. Gary Gygax and Dave Arneson. TSR. Printed rulebook (*D&D Basic Set*, 1st edition, 1977).
- Dungeon World*. Sage LaTorra and Adam Koebel. The Burning Wheel. Printed rulebook. 2015.
- Dwarf Fortress*. See “Slaves to Armok: God of Blood Chapter II: Dwarf Fortress.”
- The Elder Scrolls V: Skyrim*. Bruce Nesmith, Kurt Kuhlmann, and Emil Pagliarulo. Bethesda Game Studios/Bethesda Softworks. Windows/PlayStation 3/Xbox 360. 2011.

Elegy For a Dead World. Ichiro Lambe and Ziba Scott. Dejobaan Games. Unity. 2014.

The Empire of the Petal Throne. M. A. R. Barker. TSR. Printed rulebook. 1975.

Façade. Michael Mateas and Andrew Stern. Procedural Arts. Windows/Mac OS X. 2005.

Fallen London. Alexis Kennedy and Paul Arendt. Failbetter Games. HTML/Javascript (server). 2009 (originally as *Echo Bazaar*).

Final Fantasy II. Hiromichi Tanaka, Akitoshi Kawazu, and Koichi Ishii. Square. Famicon. 1988.

Fiasco. Jason Morningstar. Bully Pulpit Games. Printed rulebook. 2009.

Firewatch. Chris Remo, Jake Rodkin, James Benson, and Nels Anderson. Campo Santo/Limited Run Games. Windows/OS X/PlayStation 4. 2016.

Flower. Nicholas Clark and Jenova Chen. Thatgamecompany / Sony Computer Entertainment. PlayStation 3. 2009.

The Fool's Errand. Cliff Johnson. Miles Computing. Apple Macintosh. 1987.

Football Manager 2015. Sports Interactive/Sega. Windows/OS X/Linux. 2014.

The (Former) General In His Labyrinth. Mohsin Hamid. We Tell Stories/Penguin. HTML/Javascript. 2008.

Gabriel Knight 3: Blood of the Sacred, Blood of the Damned. Jane Jensen. Sierra Studios. Windows. 1999.

Gone Home. Steve Gaynor. The Fullbright Company. Windows/OS X. 2013.

Gossip. Chris Crawford. Atari. Atari 400/800. 1983.

The Gostak. Carl Muckenhoupt. Inform 6/Z. 2001.

Grim Fandango. Tim Schafer. LucasArts. Windows. 1998.

Hadean Lands. Andrew Plotkin. Zarfhome Software. Inform 7/Glulx. 2014.

Heavy Rain. David Cage. Quantic Dream/Sony Computer Entertainment. PlayStation 3. 2010.

Her Story. Sam Barlow. Windows/OS X/iOS. 2015.

Hillfolk. Robin D. Laws. Pelgrane Press. Printed rulebook. 2013.

- Hollywood Visionary*. Aaron A. Reed. Choice of Games. ChoiceScript/Multiplatform. 2015.
- Hoyle's Official Book of Games: Volume 1*. Sierra On-Line. MS-DOS/Macintosh/Amiga/Atari ST. 1989.
- The Ice-Bound Concordance*. Jacob Garbe and Aaron A. Reed. Down to the Wire. iPad/Windows. 2016.
- JUGGERNAUT*. Jason Morningstar. Bully Pulpit Games. Deck of cards. 2015.
- King of Dragon Pass*. David Dunham, Greg Stafford, and Robin D. Laws. A Sharp. PC/Mac. 1999.
- King's Quest*. Matt Korba. The Odd Gentlemen/Sierra Entertainment. Windows/-PlayStation 4/Xbox One. 2015-2016 (episodic).
- King's Quest IV: The Perils of Rosella*. Roberta Williams. Sierra On-Line. MS-DOS/Amiga/Apple II. 1988.
- L.A. Noire*. Alex Carlyle. Team Bondi/Rockstar Games. PlayStation 3/Xbox 360. 2011.
- Life in the Garden: A Deck of Stories*. Eric Zimmerman. Razorfish. Physical game with 52 loose cards and cover. 2000.
- Life is Strange*. Baptiste Moisan, Sebastien Judit, and Sebastien Gaillard. Dontnod Entertainment/Square Enix. Windows/PlayStation 3/PlayStation 4/Xbox 360/Xbox One. 2015.
- Limbo*. Jeppe Carlsen. Playdead/Microsoft Game Studios. Xbox Live Arcade. 2010.
- Long Live the Queen*. Georgina Bensley (as "Spiky Caterpillar"). Hanako Games. Ren'Py. 2012.
- Lovecraftesque*. Joshua Fox and Becky Annison. Black Armada. Printed rulebook. 2016.
- Magical Diary*. Georgina Bensley. Hanako Games. Ren'Py. 2011.
- maybe make some change*. Aaron A. Reed. Inform 7/HTML/Javascript/multimedia. 2011.
- Metroid*. Satoru Okada. Nintendo. Famicom/NES. 1986/7.
- Microscope*. Ben Robbins. Lame Mage Productions. Printed rulebook. 2011.
- Middle-earth: Shadow of Mordor*. Bob Roberts and Chris Hoge. Monolith Productions/Warner Bros. Interactive Entertainment. Windows/PlayStation 4/Xbox One. 2014.

Minecraft. Markus Persson. Mojang Specifications. Java. 2011.

Monsterhearts. Avery Alder McDaldno. Buried Without Ceremony. Printed rulebook. 2012.

Monument Valley. Ken Wong. Ustwo Games. iOS/Android. 2014.

Most Sincere Greetings, Esteemed One. Dietrich Squinkifer and Jess Marcotte. Web browser (JavaScript) / installation. 2016.

Myst. Rand Miller and Robyn Miller. Cyan/Brøderbund. Macintosh CD-ROM. 1993.

Mythic RPG. Thomas Pigeon. Word Mill Publishing. Printed rulebook. 2007.

Obduction. Rand Miller. Cyan Worlds. Windows. 2016.

Once Upon A Time. Richard Lambert, Andrew Rilstone, and James Wallis; Atlas Games. Card game. 1994.

Parable of the Polygons. Vi Hart and Nicky Case. Browser. 2014.

Poker Night at the Inventory. Telltale Games. Windows/OS X. 2010.

Polaris: Chivalric Tragedy at Utmost North. Ben Lehman. Tao Games. Printed rulebook (“Up Above the World” edition). 2005.

Pong. Allan Alcorn. Atari. Arcade cabinet. 1972.

The Prisoner in Block Nineteen. Aaron A. Reed. Javascript/Undum. 2016.

Prom Week. Josh McCoy, Mike Treanor, Ben Samuel, and Aaron Reed. Flash. 2012.

Proteus. Ed Key and David Kanaga. Windows/OS X. 2013.

Quest For Glory II: Trial by Fire. Lori Ann Cole and Corey Cole. Sierra On-Line. MS-DOS/Amiga. 1990.

Risk: The Game of Global Domination. Albert Lamorisse. Parker Brothers. Board Game. 1959.

Riven. Robyn Miller and Richard Vander Wende. Cyan/Red Orb Entertainment/Brøderbund. PC/Mac DVD-ROM. 1997.

Rogue. Michael Toy and Glenn Wichman. Unix. 1980.

The Room. Robert Dodd and Mark Hamilton. Fireproof Games. iOS. 2012.

Save the Date. Chris Cornell. Paper Dino Software. Ren’Py. 2013.

Scribblenauts. Matt Cox. 5th Cell/Warner Bros. Interactive Entertainment. Nintendo DS. 2009.

The Secret of Monkey Island. Ron Gilbert, Dave Grossman, and Tim Schafer. Lucasfilm Games. SCUMM. 1990.

The Seventh Guest. Rob Landeros and Graeme Devine. Trilobyte/Virgin Interactive. PC CD-ROM. 1993.

The Shab-al-Hiri Roach. Jason Morningstar. Bully Pulpit Games. Printed rulebook. 2006.

The Sims. Will Wright. Maxis/Electronic Arts. Windows. 2000.

The Sims 3. Ray Mazza, Matt Goss, Eric Holmberg-Weidler, and Juan Custer. The Sims Studio/Electronic Arts. Windows/OS X. 2009.

SLAMMED! Paolo Chikiamco. Choice of Games. ChoiceScript. 2014.

Slaves to Armok: God of Blood Chapter II: Dwarf Fortress. Tarn Adams and Zach Adams. Bay 12 Games. Windows/OS X/Linux. 2006-present.

Sleep is Death (Geisterfahrer). Jason Rohrer. Windows/Mac/Linux. 2010.

Sorcerer. Ron Edwards. Adept Press. Game rules. 2001.

SpaceChem. Zach Barth. Zachtronics Industries. Windows. 2011.

Space Quest II: Vohaul's Revenge. Scott Murphy and Mark Crowe. Sierra On-Line. AGI. 1987.

Spore and *Spore Creature Creator*. Will Wright. Maxis/Electronic Arts. Windows/Mac OS. 2008.

The Stanley Parable (original). Davey Wreden. Half-Life 2 mod. 2011.

The Stanley Parable. Davey Wreden and William Pugh. Galactic Cafe. Windows. 2013.

State of Decay. James Phinney, Richard Foge, and Nick Mhley. Undead Labs / Microsoft Studios. Xbox 360. 2013.

Super Mario Bros. Shigeru Miyamoto and Takashi Tezuka. Nintendo. NES. 1985.

Tactics. Charles S. Roberts. Avalon Hill. Board game. 1954.

Tales of the Crystals. Milton Bradley. Board Game. 1993.

Tetris. Alexey Pajitnov and Vladimir Pokhilko. Electronika 60/IBM PC. 1986.

Threes! Asher Vollmer. Sirvo. iOS/Android. 2014.

Trinity. Brian Moriarty. Infocom. ZIL. 1986.

Trust & Betrayal: The Legacy of Siboot. Chris Crawford. Mindscape. Macintosh. 1987.

Universalis. Ralph Mazza and Mike Holmes. Ramshead Publishing. Printed rulebook. 2002.

Untitled Journal Game. Andrew Plotkin. E BLONG, <http://www.eblong.com/zarf/uru/rj/essay-more-forum-games.html> . Game rules. 2007.

The Vanishing of Ethan Carter. Adrian Chmielarz and Krzysztof Justyński. The Astronauts. Windows. 2015.

The Warren. Marshall Miller. Bully Pulpit Games. Printed rulebook. 2015.

what if im the bad guy. Aaron A. Reed. Augmented reality installation using iPad/-Palimpsest. 2011.

What Remains of Edith Finch. Chris Bell. Giant Sparrow/Annapurna Interactive. Windows/PlayStation 4. 2017.

Winterstrike. Yoon Ha Lee. Failbetter Games. StoryNexus. 2012.

The Witness. Jonathan Blow. Thekla. PlayStation 4/Windows. 2016.

The Wolf Among Us. Ryan Kaufman, Joe Pinnery, Matt LoPresti, Aaron Casillas, John Drake and Kendall Davis. Telltale Games. Windows/Xbox 360/OS X/PlayStation 3. 2013.

Zork. Tim Anderson, Marc Blank, Dave Lebling, and Bruce Daniels. Infocom. ZIL. TRS-80/Apple II. 1980.

Bibliography

- Aarseth, E. (1997). *Cybertext : perspectives on ergodic literature*. Johns Hopkins Univ. Press, Baltimore, MD.
- Aarseth, E. (2007). Doors and perception: Fiction vs. simulation in games. *Intermedialités: Histoire et théorie des arts, des lettres et des techniques/Intermediality: History and Theory of the Arts, Literature and Technologies*, (9):35–44.
- Adams, E. (2009). The designer’s notebook: Sorting out the genre muddle. *Gamasutra*. Jul 9. http://www.gamasutra.com/view/feature/132463/the_designers_notebook_sorting_.php.
- Adams, E. (2013). *Resolutions to some problems in interactive storytelling*. Ph.D. dissertation, Univ. of Teesside.
- Agre, P. (1997). *Computation and human experience*. Cambridge Univ. Press, Cambridge.
- Antoun, C., Antoun, M., Ryan, J. O., Samuel, B., Swanson, R., and Walker, M. A. (2015). Generating natural language retellings from Prom Week play traces. *Proceedings of the FDG workshop on Procedural Content Generation in Games*.
- Appelcline, S. (2014). *Designers & dragons: A history of the roleplaying game industry*, volume The ’00s. Evil Hat Productions, Silver Spring, MD.
- Arendt, P. (2010). Echo Bazaar narrative structures. *Failbetter Games: News*. Feb. 27. blog.failbettergames.com/post/Echo-Bazaar-Narrative-Structures-part-one.aspx.
- Arinbjarnar, M., Barber, H., and Kudenko, D. (2009). A critical review of interactive drama systems. In *AISB 2009 Symposium. AI & Games, Edinburgh*.
- Ashwell, S. K. (2013). There are people going lonely. *Interactive Fiction Database: 18 Cadenca*. Apr. 7. <http://ifdb.tads.org/viewgame?id=xu3axh98nft8qm8n&review=21090>.
- Ashwell, S. K. (2016). What is a storygame? *These Heterogenous Tasks*. <https://heterogenoustasks.wordpress.com/2016/01/12/what-is-a-storygame/>.
- Baker, D. V. (2010). *Apocalypse World*. Lumpley Games.

- Bal, M. (2009). *Narratology: Introduction to the theory of narrative*. Univ. of Toronto Press, Toronto.
- Barba, R. (1997). *Riven: The sequel to Myst: The official strategy guide*. Prima Publishing.
- Barlow, S. (2016). Making ‘Her Story’—telling a story using the player’s imagination. Talk, Game Developer’s Conference.
- Beaumont, F. (1613). *The knight of the burning pestle: Full of mirth and delight*. London.
- Bentley, N. (1995). *The Ethnography of manners: Hawthorne, James, Wharton*. Cambridge Univ. Press.
- Bernstein, M. (2009). On hypertext narrative. In *Proceedings of the 20th ACM conference on Hypertext and hypermedia*, pages 5–14. ACM.
- Bernstein, M. and Greco, D. (2004). Card Shark and Thespis: Exotic tools for hypertext narrative. In Wardrip-Fruin, N. and Harrigan, P., editors, *First Person: New Media as Story, Performance, and Game*. MIT Press, Cambridge, Mass.
- Bernstein, M., Millard, D. E., and Weal, M. J. (2002). On writing sculptural hypertext. In *Proceedings of the thirteenth ACM conference on Hypertext and hypermedia*, pages 65–66. ACM.
- Bogost, I. (2017). Video games are better without stories. *The Atlantic*. Apr. 25. <https://www.theatlantic.com/technology/archive/2017/04/video-games-stories/524148/>.
- Boluk, S. and LeMieux, P. (2013). Dwarven epitaphs: Procedural histories in Dwarf Fortress. In Hayles, N. K. and Pressman, J., editors, *Comparative Textual Media: Transforming the Humanities in the Postprint Era*. Univ. of Minnesota Press.
- Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, 99(suppl 3):7280–7287.
- Boss, E. C. (2008). Key concepts in Forge Theory. In Montola, M. and Stenros, J., editors, *Playground Worlds*. Ropecon ry.
- Boyd, B. (1999). *Nabokov’s Pale Fire: The magic of artistic discovery*. Princeton Univ. Press, Princeton, NJ.
- Bryan, J. S. (2013). *The dynamics of the player narrative: How choice shapes videogame literature*. Ph.D. dissertation, Georgia Institute of Technology.
- Buckles, M. A. (1985). *Interactive fiction: The computer storygame Adventure*. Ph.D. dissertation, Univ. of California San Diego.

- Bura, S. (2012). Emotional AI for expanding worlds. In Panel session “Beyond Eliza: Constructing Socially Engaging AI,” Game Developers Conference.
- Burroughs, W. S. (1961). The soft machine. *Grove*.
- Campbell, C. (2016). The problem with ‘walking sims’. *Polygon*. Sep. 28. <https://www.polygon.com/2016/9/28/13076654/the-problem-with-walking-sims>.
- Charles, D. and Black, M. (2004). Dynamic player modeling: A framework for player-centered digital games. In *Proc. of the International Conference on Computer Games: Artificial Intelligence, Design and Education*, pages 29–35.
- Chemers, M. M. (2010). *Ghost Light: An Introductory Handbook for Dramaturgy*. Southern Illinois Univ. Press.
- Chemers, M. M. (2013). ‘Lyke unto a lively thing’: Theatre history and social robotics. In *Theatre, Performance and Analogue Technology*, pages 232–249. Springer.
- Clancey, W. J. (1983). The epistemology of a rule-based expert system—A framework for explanation. *Artificial intelligence*, 20(3):215–251.
- Clark, H. H., Brennan, S. E., et al. (1991). Grounding in communication. *Perspectives on socially shared cognition*, 13(1991):127–149.
- Collins, B. (2016). How to make sure your robot doesn’t become a Nazi. *The Daily Beast*. Mar. 27. <http://www.thedailybeast.com/articles/2016/03/27/how-to-make-sure-your-robot-doesn-t-become-a-nazi>.
- Compton, K. (2016). So you want to build a generator... Feb. 22. <http://galaxykate0.tumblr.com/post/139774965871/so-you-want-to-build-a-generator>.
- Compton, K., Kybartas, B., and Mateas, M. (2015). Tracery: An author-focused generative text tool. In *International Conference on Interactive Digital Storytelling*, pages 154–161. Springer.
- Conway, J. (1970). The game of life. *Scientific American*, 223(4):4.
- Cork, J. (2016). Firewatch: Gabbin’ in the woods. *Game Informer*. Feb 8.
- Cortázar, J. (1966). *Hopscotch (Rayuela)*. Pantheon Books, New York. Translated from the Spanish by Gregory Rabassa.
- Costikyan, G. (2005). Game styles, innovation, and new audiences: An historical view. *Digital Games Research Conference*.
- Crawford, C. (1987). Process intensity. *The Journal of Computer Game Development*, 1(5).

- Crawford, C. (1999). Assumptions underlying the Erasmatron interactive storytelling engine. In *AAAI Fall Symposium on Narrative Intelligence*, pages 112–114.
- Crawford, C. (2007). Deikto: A language for interactive storytelling. In Harrigan, P. and Wardrip-Fruin, N., editors, *Second Person*. The MIT Press, Cambridge, Mass.
- Crawford, C. (2012). *Chris Crawford on interactive storytelling*. New Riders/Peachpit, San Francisco, 2nd edition.
- Csikszentmihalyi, M. (1996). *Creativity: Flow and the psychology of discovery and invention*. New York.
- Daly, L. (2016). Chatbot fundamentals: An interactive guide to writing bots in Python. <https://apps.worldwritable.com/tutorials/chatbot/>.
- de Quidt, J. (2016). Wot I Think: The Ice-Bound Concordance. *Rock Paper Shotgun*. Mar. 2. <https://www.rockpapershotgun.com/2016/03/02/the-ice-bound-concordance-review/>.
- Donlan, C. (2016). Reading and writing in Ice-Bound, an unmissable narrative game. *Eurogamer*. Feb. 21. <http://www.eurogamer.net/articles/2016-02-21-reading-and-writing-in-ice-bound-an-unmissable-narrative-game>.
- Doob, P. R. (1990). *The idea of the labyrinth from classical antiquity through the Middle Ages*. Cornell Univ. Press.
- Doorenbos, R. B. (1995). *Production matching for large learning systems*. Ph.D. dissertation, Univ. of Southern California.
- Dormans, J. et al. (2012). *Engineering emergence: Applied theory for game design*. Creative Commons.
- Douglass, J. (2007). *Command lines: Aesthetics and technique in interactive fiction and new media*. Ph.D. dissertation, Univ. of California Santa Barbara.
- Drachen, A. and Smith, J. H. (2008). Player talk—The functions of communication in multiplayer role-playing games. *Computers in Entertainment (CIE)*, 6(4):56.
- Dunham, D. (2011). The problem with branching. *Failbetter Games Blog*. Oct. 4. <http://www.failbettergames.com/the-problem-with-branching/>.
- Dyer, J. (2011). Repetition of text in interactive fiction. In Jackson-Mead, K. and Wheeler, J. R., editors, *IF Theory Reader*. >TRANSCRIPT ON PRESS.
- Dyer, J. (2014). Review: Blood and Laurels. June 12, 2014. <https://bluerenga.wordpress.com/2014/06/12/review-blood-and-laurels/>.

- Ebert, R. (2005). Why did the chicken cross the genders? *Movie Answer Man*. Nov. 27. <http://www.rogerebert.com/answer-man/why-did-the-chicken-cross-the-genders>.
- Ebert, R. (2010a). Video games can never be art. *Roger Ebert's Journal*. Apr. 16. <http://www.rogerebert.com/rogers-journal/video-games-can-never-be-art>.
- Ebert, R. (2010b). Okay, kids, play on my lawn. *Roger Ebert's Journal*. July 1. <http://www.rogerebert.com/rogers-journal/okay-kids-play-on-my-lawn>.
- Edwards, R. (2001a). GNS and other matters of roleplaying theory. *The Forge*, 14.
- Edwards, R. (2001b). *Sorcerer*. Adept Press.
- Edwards, R. (2003). Narrativism: Story now. *The Forge*. http://www.indie-rpgs.com/_articles/narr_essay.html.
- Egri, L. (1960). *The Art of Dramatic Writing*. Westview, Boulder.
- Ellison, C. (2013). Anna Anthropy and the Twine revolution. *The Guardian*.
- Ensslin, A. (2014). Of windsighs and wayfaring: *Blue Lacuna*, an epic interactive fiction. In *Literary Gaming*, pages 105–122. MIT Press.
- Eskelinen, M. (2001). Towards computer game studies. *Digital creativity*, 12(3):175–183.
- Evans, R. and Short, E. (2014). Versu—A simulationist storytelling system. *IEEE Transactions on Computational Intelligence and AI in Games*, 6(2):113–130.
- Fabulich, D. (2011). By the numbers: How to write a long interactive novel that doesn't suck. Jul. 27. <https://www.choiceofgames.com/2011/07/by-the-numbers-how-to-write-a-long-interactive-novel-that-doesnt-suck/>.
- Ferguson, W., Diller, D., Moffitt, K., Roberts, B., Coon, W., Mateas, M., Shapiro, D., Stern, A., LeBron, L., and Tannenbaum, K. (2015). IMMERSE: Interactive mentoring for multimodal experiences in realistic social encounters. Technical report, DTIC Document.
- Fernández-Vara, C. (2009). *The tribulations of adventure games: integrating story into simulation through performance*. Ph.D. dissertation, Georgia Institute of Technology.
- Fernández-Vara, C. (2011). From “open mailbox” to context mechanics: Shifting levels of abstraction in adventure games. In *Proceedings of the 6th International Conference on Foundations of Digital Games*, pages 131–138. ACM.
- Fernández-Vara, C. and Thomson, A. (2012). Procedural generation of narrative puzzles in adventure games: The puzzle-dice system. In *Proceedings of the the Third Workshop on Procedural Content Generation in Games*, page 12. ACM.

- Fox, J. and Annison, B. (2016). *Lovecraftesque*. Black Armada.
- Fuller, M. (2008). *Software studies: A lexicon*. MIT Press.
- Garbe, J., Reed, A. A., Dickinson, M., Wardrip-Fruin, N., and Mateas, M. (2014). Author-assist visualizations for Ice-Bound, a combinatorial narrative. *Foundations of Digital Games*.
- Garcin, P. (2013). Narrative generators: Sports games and story engines. *Talk, Game Developer’s Conference*.
- Gaskill, J. (2009). Quantic Dream’s David Cage: Play Heavy Rain several times, “kill the magic of it”. *G4 TV*. Aug. 31. <http://www.g4tv.com/thefeed/blog/post/698809/quantic-dreams-david-cage-play-heavy-rain-several-times-kill-the-magic-of-it/>, accessed 3-15-2012.
- Gilbert, R. (2004). Why adventure games suck. May 13; original version 1989. http://grumpygamer.com/why_adventure_games_suck.
- Gingold, C. (2016). *Play Design*. Ph.D. dissertation, Univ. of California, Santa Cruz.
- Greene, J. (2016). Microsoft muzzles its artificially intelligent Twitter persona. *The Wall Street Journal*. Mar. 24.
- Grip, T. (2017). Traversal and the problem with walking simulators. *Gamasutra*. Mar. 23. http://www.gamasutra.com/blogs/ThomasGrip/20170323/294377/Traversal_and_the_Problem_with_Walking_Simulators.php.
- guidetofootballmanager.com (2015). Guide to football manager. accessed 6-24-16.
- Gygax, G. (1979). *Advanced Dungeons & Dragons: Dungeon Master’s Guide*. TSR Games, 1st edition.
- Hamlet, R. (1994). Random testing. In Marciniak, J. J., editor, *Encyclopedia of Software Engineering*. John Wiley, New York.
- Haraway, D. J. (2008). *When species meet*. Univ. of Minnesota Press.
- Haraway, D. J. (2016). *Staying with the trouble: Making kin in the Chthulucene*. Duke Univ. Press.
- Harrell, D. F. (2007). GRIOT’s Tales of Haints and Seraphs: A computational narrative generation system. In Harrigan, P. and Wardrip-Fruin, N., editors, *Second Person*. MIT Press, Cambridge, Mass.
- Hartnell, T. (1984). *Creating Adventure Games on Your Computer*. Ballantine.
- Harviainen, J. T. (2009). A hermeneutical approach to role-playing analysis. *International Journal of Role-Playing*, 1(1):66–78.

- Henry, L. (2003). Group narration: Power, information, and play in role playing games. *Darkshire.net*.
- Higgins, D. (1987). *Pattern poetry : guide to an unknown literature*. State Univ. of New York Press, Albany.
- Hill, K. (2015). After Twitter bot makes death threat, its owner gets questioned by police. *Fusion*. Feb. 11. <https://fusion.kinja.com/after-twitter-bot-makes-death-threat-its-owner-gets-qu-1793845332>.
- Hirst, D. L. (1979). *Comedy of manners*, volume 40 of *The Critical Idiom*. Methuen & Co Ltd.
- Hooper, C. J. and Weal, M. J. (2005). The StorySpinner sculptural reader. In *Proceedings of the Sixteenth ACM Conference on Hypertext and Hypermedia*, pages 288–289. ACM.
- Huizinga, J. (1950). *Homo ludens: A study of the play-element in culture*. Roy Publishers.
- Hunicke, R. (2005). The case for dynamic difficulty adjustment in games. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 429–433. ACM.
- Hunicke, R., LeBlanc, M., and Zubek, R. (2004). MDA: A formal approach to game design and game research. In *Proceedings of the AAAI Workshop on Challenges in Game AI*, volume 4, page 1.
- Jayanth, M. (2016). Forget protagonists: Writing NPCs with agency for ‘80 Days’ and beyond. Talk, Game Developer’s Conference.
- Jeong, S. (2016). How to make a bot that isn’t racist. *Vice Motherboard*. Mar. 25. https://motherboard.vice.com/en_us/article/how-to-make-a-not-racist-bot.
- Johnson, B. S. (1969). *The unfortunates*. Panther in association with Secker & Warburg, London.
- Johnston, C. (2013). Dwarf Fortress: Ten hours with the most inscrutable video game of all time. *Ars Technica*. Feb. 24. <https://arstechnica.com/gaming/2013/02/dwarf-fortress-ten-hours-with-the-most-inscrutable-video-game-of-all-time/>.
- Johnstone, K. (2012). *Impro: Improvisation and the theatre*. Routledge.
- Jordà, S., Kaltenbrunner, M., Geiger, G., and Bencina, R. (2005). The Reactable. In *Proceedings of the international computer music conference (ICMC 2005), Barcelona, Spain*, pages 579–582. Citeseer.
- Joyce, M. (1991). Storyspace as a hypertext system for writers and readers of varying ability. In *Proceedings of the third annual ACM conference on Hypertext*, pages 381–387. ACM.

- Juul, J. (2005). *Half-Real: Video games between real rules and fictional worlds*. MIT Press, Cambridge, Mass.
- Juul, J. (2008). Fear of failing? the many meanings of difficulty in video games. In Perron, B. and Wolf, M. J., editors, *The Video Game Theory Reader 2*, pages 237–252. Routledge.
- Kaiser, R. (2013). The surprising design of Crusader Kings II. *Gamasutra*. Jan. 6 2013. http://www.gamasutra.com/view/feature/184356/the_surprising_design_of_crusader_.php.
- Kennedy, A. (2012). StoryNexus developer diary #1: ‘Worlds exist’. *Failbetter Games Blog*. June 29. <https://www.failbettergames.com/storynexus-developer-diary-1-worlds-exist/>.
- Kennedy, A. (2016). Choice, consequence and complicity. Talk, Game Developer’s Conference.
- Kent, A. and Williams, J. G. (1989). *Encyclopedia Of Microcomputers*. CRC Press.
- Khandaker-Kokoris, M. (2015). Thinking about people: Designing games for social simulation. Talk, Game Developer’s Conference.
- Kidder, T. (1981). *The soul of a new machine*. Little, Brown and Company, New York.
- Kleinman, E. M. (2016). Understanding metagaming mechanics in interactive storytelling. Master’s thesis, Drexel University.
- Klimas, C. (2009). Twine. <https://twinery.org/>.
- Koenitz, H. (2010). Towards a theoretical framework for interactive digital narrative. In *Joint International Conference on Interactive Digital Storytelling*, pages 176–185. Springer.
- Koenitz, H. (2011). Extensible tools for practical experiments in IDN: The Advanced Stories Authoring and Presentation System. In *International Conference on Interactive Digital Storytelling*, pages 79–84. Springer.
- Konzack, L. (2015). The Wunderkammer-Gesamtkunstwerk model: A framework for role-playing game analysis and design. Directly published on Academia.edu.
- Koster, R. (2013). *A Theory of fun for game design*. O’Reilly Media.
- Kuipers, B., McCarthy, J., and Weizenbaum, J. (1976). Computer power and human reason. *ACM SIGART Bulletin*, (58):4–13.
- Landow, G. P. (1994). What’s a critic to do?: Critical theory in the age of hypertext. In Landow, G. P., editor, *Hyper/Text/Theory*. Johns Hopkins Univ. Press.

- Landow, G. P. (2009). Reconfiguring writing. In Bernstein, M. and Greco, D., editors, *Reading Hypertext*. Eastgate Systems.
- Laurel, B. (2013). *Computers as theatre*. Addison-Wesley, 2nd edition.
- Laurel, B. K. (1986). *Toward the design of a computer-based interactive fantasy system*. Ph.D. dissertation, The Ohio State University.
- Laws, R. (1995). The hidden art: Slouching towards a critical framework for RPGs. *Inter*Action*, (#1).
- Laws, R. D. (2010). *Hamlet's Hit Points*. Gameplaywright Press.
- Le Guin, U. K. (1968). A wizard of Earthsea. *Parnassus*.
- Lebling, P. D., Blank, M. S., and Anderson, T. A. (1979). Zork: A computerized fantasy simulation game. *IEEE Computer*, 12(4):51–59.
- Lebowitz, M. (1984). Creating characters in a story-telling universe. *Poetics*, 13(3):171–194.
- Lehman, B. (2005). *Polaris: Chivalric tragedy at utmost north*. Tao Games.
- Levy, G. (2005). *112 acting games: A comprehensive workbook of theatre games for developing acting skills*. Meriwether Publishing.
- Livo, N. J. and Rietz, S. A. (1986). *Storytelling: Process and practice*. Libraries Unlimited, Littleton, Colorado.
- Lowe, A. (1999). The death of adventure games. Mar. 19. <http://allowe.com/al/articles/death-of-adventures.html>.
- Mahardy, M. (2016). Doors of perception. *Gamespot*. Jan. 25. <https://www.gamespot.com/reviews/the-witness-review/1900-6416336/>.
- Maher, J. (2008). Toward games that matter: The promise and problems of the storygame. <http://maher.filfre.net/if/storygames.htm>.
- Maher, J. (2011). A busy 1979. *The Digital Antiquarian*. <http://www.filfre.net/2011/07/a-busy-1979/>.
- Marshall, C. C. and Shipman III, F. M. (1995). Spatial hypertext: Designing for change. *Communications of the ACM*, 38(8):88–97.
- Mateas, M. (2004). A preliminary poetics for interactive drama and games. In Wardrip-Fruin, N. and Harrigan, P., editors, *First Person: New Media as Story, Performance, and Game*. MIT Press, Cambridge, Mass.

- Mateas, M. and McCoy, J. (2014). An architecture for character-rich social simulation. In Rabin, S., editor, *Game AI Pro: Collected Wisdom of Game AI Professionals*, pages 515–30. CRC Press.
- Mateas, M. and Stern, A. (2003). Façade: An experiment in building a fully-realized interactive drama. Talk and Proceedings, Game Developers Conference.
- Mateas, M. and Stern, A. (2005). Procedural authorship: A case-study of the interactive drama Façade. *Digital Arts and Culture (DAC)*.
- Mateas, M. and Stern, A. (2007). Writing Façade: A case study in procedural authorship. In Harrigan, P. and Wardrip-Fruin, N., editors, *Second Person*. MIT Press, Cambridge, Mass.
- Mateas, M. and Wardrip-Fruin, N. (2009). Defining operational logics. *Breaking New Ground: Innovation in Games, Play, Practice and Theory; Proceedings of DiGRA 2009*.
- Mawhorter, P. (2016). *Artificial intelligence as a tool for understanding narrative choices*. PhD dissertation, Univ. of California Santa Cruz.
- Mawhorter, P., Mateas, M., Wardrip-Fruin, N., and Jhala, A. (2014). Towards a theory of choice poetics. *Foundations of Digital Games*.
- McCaffrey, T. and Spector, L. (2011). AhaNets: Structuring semantic networks to enhance innovation. *ACM Creativity & Cognition Workshop on Semi-Automated Creativity*.
- McCartney, S. (1999). *ENIAC: The triumphs and tragedies of the world’s first computer*. Walker & Company.
- McCloud, S. (1994). *Understanding comics*. HarperCollins.
- McCoy, J. (2012). *All the world’s a stage: A playable model of social interaction inspired by dramaturgical analysis*. Ph.D. dissertation, Univ. of California Santa Cruz.
- McCoy, J., Treanor, M., Samuel, B., Reed, A. A., Mateas, M., and Wardrip-Fruin, N. (2013). Prom Week: Designing past the game/story dilemma. In *Foundations of Digital Games*.
- McCoy, J., Treanor, M., Samuel, B., Reed, A. A., Wardrip-Fruin, N., and Mateas, M. (2014). Social story worlds with Comme il Faut. *Transactions on Computational Intelligence and AI in Games Special Issue on Computational Narrative and Games*, 6(2):97–112.
- Meehan, J. R. (1977). TALE-SPIN, an interactive program that writes stories. In *International Joint Conference on Artificial Intelligence*, volume 77, pages 91–98.

- Mehm, F., Göbel, S., and Steinmetz, R. (2013). An authoring tool for educational adventure games: Concept, game models and authoring processes. *International Journal of Game-Based Learning (IJGBL)*, 3(1):63–79.
- Miéville, C. (2009). China Miéville on crime novels. *Whatever*, (May 26). Scalzi, J., editor. <http://whatever.scalzi.com/2009/05/26/china-mieville-on-crime-novels/>.
- Mitchell, A. and McGee, K. (2012). Reading again for the first time: A model of rereading in interactive stories. In *International Conference on Interactive Digital Storytelling*, pages 202–213. Springer.
- Montfort, N. (2003). *Twisty little passages : an approach to interactive fiction*. MIT Press, Cambridge, Mass.
- Montfort, N. (2004). Interactive fiction as ‘story,’ ‘game,’ ‘storygame,’ ‘novel,’ ‘world,’ ‘literature,’ ‘puzzle,’ ‘problem,’ ‘riddle,’ and ‘machine’. In Wardrip-Fruin, N. and Harrigan, P., editors, *First person: New media as story, performance, and game*, pages 310–317. MIT Press, Cambridge, Mass.
- Montfort, N. (2009). Curveship: An interactive fiction system for interactive narrating. *NAACL HLT Workshop on Computational Approaches to Linguistic Creativity*.
- Montfort, N., Pérez, R., Harrell, D. F., and Campana, A. (2013). Slant: A blackboard system to generate plot, figuration, and narrative discourse aspects of stories. In *Proceedings of the fourth international conference on computational creativity*, pages 168–175.
- Morningstar, J. (2009). *Fiasco*. Bully Pulpit Games.
- Motte, W. F. (1998). *Oulipo: A primer of potential literature*. French literature series. Dalkey Archive Press, Normal, Ill., 1st dalkey archive edition.
- Muncy, J. (2016). Now that I beat The Witness, I see what’s wrong with it. *Wired*. Feb. 11. <https://www.wired.com/2016/02/the-witness/>.
- Murray, J. (1997). *Hamlet on the holodeck: The future of narrative in cyberspace*. Free Press, New York.
- Murray, J. (2004). From game-story to cyberdrama. In Wardrip-Fruin, N. and Harrigan, P., editors, *First person: New media as story, performance, and game*. MIT Press, Cambridge, Mass.
- Murray, J. (2011). Why Paris needs Hector and Lancelot needs Mordred: Using traditional narrative roles and functions for dramatic compression in interactive narrative. In *Fourth International Conference on Interactive Digital Storytelling*.
- Nabokov, V. V. (1962). *Pale fire; a novel*. Putnam, New York.

- Nelson, G. (1995). The craft of adventure. <http://www.ifarchive.org/if-archive/info/Craft.Of.Adventure.txt>.
- Nelson, G. (2014a). Prompter: A domain-specific language for Versu. https://versublog.files.wordpress.com/2014/05/graham_versu.pdf.
- Nelson, P. (2014b). Blood and Laurels review. Dec. 11. <https://thestoryelement.wordpress.com/2014/12/11/blood-and-laurels-review/>.
- Orkin, J. and Roy, D. (2007). The Restaurant Game: Learning social behavior and language from thousands of players online. *Journal of Game Development*, 3(1):39–60.
- Osborn, J., Samuel, B., Mateas, M., and Wardrip-Fruin, N. (2015). Playspecs: Regular expressions for game play traces. *Proceedings of the AIIDE*.
- Packard, E. (1979). *Choose Your Own Adventure #1: The Cave of Time*. Bantam Books, New York.
- Palumbo, A. (2016). Firewatch interview—A walkie-talkie mystery. *WCCF Tech*. Jan 9. <http://wccftech.com/firewatch-interview-walkietalkie-mystery/>.
- Parakenings, K. (2010). Blue Lacuna (review). *Jay Is Games*. URL: http://jayisgames.com/archives/2010/06/blue_lacuna.php; accessed 2-26-14.
- Pearson, C. (2012). Impressions: Prom Week. *Rock Paper Shotgun*. Feb. 26. <https://www.rockpapershotgun.com/2012/02/16/impressions-prom-week/>.
- Peinado, F. and Gervás, P. (2004). Transferring game mastering laws to interactive digital storytelling. In *International Conference on Technologies for Interactive Digital Storytelling and Entertainment*, pages 48–54. Springer.
- Pérez, R. and Sharples, M. (2001). MEXICA: A computer model of a cognitive account of creative writing. *Journal of Experimental & Theoretical Artificial Intelligence*, 13(2):119–139.
- Peterson, J. (2012). *Playing at the world: A history of simulating wars, people and fantastic adventures, from chess to role-playing games*. Unreason Press.
- Pigeon, T. (2006). *Mythic Game Master Emulator*. Word Mill Publishing.
- Pizzi, D. and Cavazza, M. (2008). From debugging to authoring: Adapting productivity tools to narrative content description. In *Joint International Conference on Interactive Digital Storytelling*, pages 285–296. Springer.
- Plotkin, A. (2011). Characterizing, if not defining, interactive fiction. In Jackson-Mead, K. and Wheeler, J. R., editors, *IF Theory Reader*. >TRANSCRIPT ON PRESS.

- Plotkin, A. (2016). Point-of-view in The Witness: Design ruminations. *The Gameshelf*. Jul. 27. <http://gameshelf.jmac.org/2016/07/point-of-view-in-the-witness-design-ruminations/>.
- Queneau, R. (1961). *Cent mille milliards de poèmes*. Gallimard.
- Rad, C. (2016). The Witness review. *IGN*. Jan. 25. <http://www.ign.com/articles/2016/01/25/the-witness-review>.
- Rand, A. (1936). *Night of January 16th*. Longmans, Green, New York.
- Rawson, P. S. (1997). *Sculpture*. Univ. of Pennsylvania Press, Philadelphia.
- Reed, A., Wardrip-Fruin, N., and Mateas, M. (2014a). The Eurekon: A design pattern in expressive storygames. In *Seventh Intelligent Narrative Technologies Workshop*.
- Reed, A. A. (2010). Blue Lacuna: Lessons learned writing the world’s longest interactive fiction. In *Electronic Literature Organization*, Providence, RI.
- Reed, A. A. (2011a). “Games” for grief, mourning, and anger. *Expressive Intelligence Studio Blog*. Oct. 7. <https://eis-blog.soe.ucsc.edu/2011/10/games-for-grief-mourning-and-anger/>.
- Reed, A. A. (2011b). A sequence of possibilities: Constructive fictions, quantum authoring, and the search for an ideal story system. Master’s thesis, Univ. of California Santa Cruz.
- Reed, A. A. (2012a). Prom Week authoring: Crafting procedurally-driven narratives. *Expressive Intelligence Studio Blog*. Feb. 9. <http://promweek.soe.ucsc.edu/2012/02/09/prom-week-authoring-crafting-procedurally-driven-narratives/>.
- Reed, A. A. (2012b). Sharing authoring with algorithms: Procedural generation of satellite sentences in text-based interactive stories. In *Workshop on Procedural Content Generation, Foundations of Digital Games*, Raleigh.
- Reed, A. A. (2012c). Where is the MS Word of interactive narrative? Talk, Electronic Literature Organization: Electrifying Literature.
- Reed, A. A. (2013). Re-visualizing Afghanistan in “what if im the bad guy”: Using Palimpsest to create an AR documentary. *Leonardo Electronic Almanac: Not Here Not There*, 19(1):158–169.
- Reed, A. A. (2015). Hollywood Visionary and gender. >*TILT AT WIND-MILLS*. July 24. <https://lacunagame.blogspot.com/2015/07/hollywood-visionary-and-gender.html>.
- Reed, A. A. and Garbe, J. (2016). The only way out is through: Combinatorial narrative. *Ice-Bound News*. http://ice-bound.com/news/combinatorial_narrative/.

- Reed, A. A., Garbe, J., Wardrip-Fruin, N., and Mateas, M. (2014b). Ice-Bound: Combining richly-realized story with expressive gameplay. In *Foundations of Digital Games*.
- Richardson, L. (2013). Bots should punch up. Nov. 27. <https://www.crummy.com/2013/11/27/0>.
- Riedl, M. O. and Young, R. M. (2004). An intent-driven planner for multi-agent story generation. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 186–193. IEEE Computer Society.
- Robbins, B. (2011). *Microscope*. Lame Mage Productions.
- Robbins, B. (2015). *Microscope Explorer*. Lame Mage Productions.
- Roberts, D. L. and Isbell, C. L. (2008). A survey and qualitative analysis of recent advances in drama management. *International Transactions on Systems Science and Applications, Special Issue on Agent Based Systems for Human Learning*, 4(2):61–75.
- Rodari, G. (1996). *The grammar of fantasy: An introduction to the art of inventing stories*. Teachers & Writers Collaborative. Trans. by Jack Zipes.
- Rogers, S. (2009). Everything I learned about level design I learned from Disneyland. Talk, Game Developer’s Conference.
- Rollings, A. and Adams, E. (2003). *Andrew Rollings and Ernest Adams on Game Design*. New Riders.
- Ryan, J., Mateas, M., and Wardrip-Fruin, N. (2016). Characters who speak their minds: Dialogue generation in Talk of the Town. *Proceedings of AIIDE*.
- Ryan, M.-L. (2001a). Beyond myth and metaphor: The case of narrative in digital media. *Game Studies*, 1.
- Ryan, M.-L. (2001b). *Narrative as virtual reality: Immersion and interactivity in literature and electronic media*. Johns Hopkins Univ. Press.
- Salter, A. (2014). *What is your quest? From adventure games to interactive books*. Univ. of Iowa Press.
- Samuel, B. (2016). *Crafting Stories Through Play*. Ph.D. dissertation, Univ. of California Santa Cruz.
- Samuel, B., Lederle-Ensign, D., Treanor, M., Wardrip-Fruin, N., McCoy, J., Reed, A., and Mateas, M. (2015a). Playing the worlds of Prom Week. *Narrative Theory, Literature, and New Media: Narrative Minds and Virtual Worlds*, pages 87–105.

- Samuel, B., McCoy, J., Treanor, M., Reed, A. A., Mateas, M., and Wardrip-Fruin, N. (2014). Introducing story sampling: Preliminary results of a new interactive narrative evaluation technique. In *Foundations of Digital Games*.
- Samuel, B., Reed, A. A., Maddaloni, P., Mateas, M., and Wardrip-Fruin, N. (2015b). The Ensemble Engine: Next-generation social physics. In *Proceedings of the Tenth International Conference on the Foundations of Digital Games (FDG 2015)*, pages 22–25.
- Saporta, M. and Howard, R. (1963). *Composition no. 1, a novel*. Simon and Schuster, New York. Translated from the French by Richard Howard.
- Sarkar, S. (2013). Redshirt dev apologizes about not warning of game’s harassment mechanic. *Polygon*. Nov. 22. <https://www.polygon.com/2013/11/22/5134172/redshirt-developer-apologizes-warning-in-game-harassment>.
- Schuette, K. (1985). *The Book of Adventure Games*. Arrays, Inc.
- Short, E. (2009). Conversation. *Emily Short’s Interactive Storytelling*. <https://emshort.blog/how-to-play/writing-if/my-articles/conversation/>.
- Short, E. (2011). NPC conversation systems. In Jackson-Mead, K. and Wheeler, J. R., editors, *IF Theory Reader*. >TRANSCRIPT ON PRESS.
- Short, E. (2015a). *The Annals of the Parrigues*. Self-published PDF. <https://drive.google.com/file/d/0B97d5C256qbr0HFwSUhsZE4tU0k/view>.
- Short, E. (2015b). Narrative and simulation in interactive dialogue. Keynote, International Conference on Interactive Digital Storytelling (ICIDS).
- Skorupski, J., Jayapalan, L., Marquez, S., and Mateas, M. (2007). Wide Ruled: A friendly interface to author-goal based story generation. In *International Conference on Virtual Storytelling*, pages 26–37. Springer.
- Smith, A. (2012). Wot I think: Crusader Kings II. *Rock Paper Shotgun*. Feb. 23. <https://www.rockpapershotgun.com/2012/02/23/wot-i-think-crusader-kings-ii/>.
- Smith, G. (2016). A proceduralist view on diversity in games. *Journal of Games Criticism*, 3(A).
- Smith, G., Whitehead, J., and Mateas, M. (2010). Tanagra: A mixed-initiative level design tool. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games*, pages 209–216. ACM.
- Stephens, A. (2012). Prom Week [review]. *alastairstephens.com*. Feb. 17. <http://alastairstephens.com/prom-night/>.
- Stephenson, N. (1995). *The Diamond Age*. Bantam Spectra.

- Suchman, L. A. (1987). *Plans and situated actions: The problem of human-machine communication*. Cambridge Univ. Press.
- Sun, R. (2006). *Cognition and multi-agent interaction: From cognitive modeling to social simulation*. Cambridge Univ. Press.
- Swanson, R. and Gordon, A. S. (2008). Say Anything: A massively collaborative open domain story writing companion. In *Joint International Conference on Interactive Digital Storytelling*, pages 32–40. Springer.
- Swanson, R. and Gordon, A. S. (2012). Say Anything: Using textual case-based reasoning to enable open-domain interactive storytelling. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2(3):16.
- Swift, J. (1726). *Gulliver’s travels into several remote nations of the world*. Project Gutenberg edition.
- Tanenbaum, J. (2008). Believability, adaptivity, and performativity: Three lenses for the analysis of interactive storytelling. Master’s thesis, Simon Fraser University.
- Tearse, B., Mateas, M., and Wardrip-Fruin, N. (2010). MINSTREL remixed: A rational reconstruction. In *Proceedings of the Intelligent Narrative Technologies III Workshop*, page 12. ACM.
- Tearse, B., Mawhorter, P., Mateas, M., and Wardrip-Fruin, N. (2014). Skald: MINSTREL reconstructed. *IEEE Transactions on Computational Intelligence and AI in Games*, 6(2):156–165.
- Terna, P. et al. (1998). Simulation tools for social scientists: Building agent based models with SWARM. *Journal of Artificial Societies and Social Simulation*, 1(2):1–12.
- Toffler, A. (1990). *Future shock*. Bantam.
- Tompkins, J. P., editor (1980). *Reader-response criticism: From formalism to post-structuralism*. John Hopkins Univ. Press, Baltimore.
- Torner, E. (2012). Immersive gameplay—Interview with Emily Care Boss. <https://guyintheblackhat.wordpress.com/2012/07/11/immersive-gameplay-interview-with-emily-care-boss/>.
- Totilo, S. (2016). The Witness: The Kotaku review. *Kotaku*. Jan. 25. <https://kotaku.com/the-witness-the-kotaku-review-1754919626>.
- Treanor, M. (2013). *Investigating procedural expression and interpretation in videogames*. Ph.D. dissertation, UC Santa Cruz.
- Treanor, M. and Mateas, M. (2011). BurgerTime: A proceduralist investigation. In *Conference of the Digital Games Research Association-DIGRA 2011*.

- Treanor, M., McCoy, J., and Sullivan, A. (2016). A framework for playable social dialogue. In *Twelfth Artificial Intelligence and Interactive Digital Entertainment Conference*.
- Turner, S. R. (1993). *MINSTREL: A computer model of creativity and storytelling*. Ph.D. dissertation.
- Turner, V. W. (1982). *From ritual to theatre: The human seriousness of play*. Performing Arts Journal Publications.
- Univ. of Richmond Writing Center (2010). A list of important literary terms. Writer's Web, <http://writing2.richmond.edu/writing/wweb/terms.html>; accessed 2011-06-08.
- Walker, A. (2017). Stories in games aren't problems, they're solutions. *Waypoint*. Apr. 26. https://waypoint.vice.com/en_us/article/stories-in-games-arent-problems-theyre-solutions.
- Walker, J. (1999). Piecing together and tearing apart: finding the story in *afternoon*. In *Proceedings of the Tenth ACM Conference on Hypertext and Hypermedia*, pages 111–117. ACM.
- Wardrip-Fruin, N. (2005). Playable media and textual instruments. *Dichtung Digital*, vol. 1. <http://www.dichtung-digital.de/2005/1/Wardrip-Fruin/index.htm>.
- Wardrip-Fruin, N. (2009). *Expressive processing: Digital fictions, computer games, and software studies*. Software studies. MIT Press, Cambridge, Mass.
- Wardrip-Fruin, N. (2010). Beyond the complex surface. *Beyond the Screen: Transformations of Literary Structures, Interfaces and Genres*, 44:227.
- Wardrip-Fruin, N., Mateas, M., Dow, S., and Sali, S. (2009). Agency reconsidered. In *Proceedings of the Digital Games Research Association Conference*.
- Ware, S. G. and Young, R. M. (2011). CPOCL: A narrative planner supporting conflict. In *AIIDE*.
- Wellhousen, K. and Kieff, J. E. (2001). *A constructivist approach to block play in early childhood*. Cengage Learning.
- Weyhrauch, P. (1997). *Guiding interactive drama*. Ph.D. dissertation, Carnegie Mellon University.
- White, O. (2016). Firewatch took away our ability to be good people, and that's where it shines. *Polygon*. Feb. 12. <https://www.polygon.com/2016/2/12/10966494/firewatch-agency-campo-santo>.
- Wolpaw, E. (2000). Who killed adventure games? *Old Man Murray*. Sep. 11. <http://www.oldmanmurray.com/features/77.html>.

- Woolley, S., boyd, d., Broussard, M., et al. (2016). How to think about bots. *Points*. Feb. 24. <https://points.datasociety.net/how-to-think-about-bots-1ccb6c396326>.
- Xuanming, Z. (2011). Designing games to be retold. Master's thesis, National Univ. of Singapore.
- Zak S (2011). *Vornheim: The Complete City Kit*. Lamentations of the Flame Princess.
- Zhu, J. and Harrell, D. F. (2008). Daydreaming with intention: Scalable blending-based imagining and agency in generative interactive narrative. In *AAAI Spring Symposium: Creative Intelligent Systems*, volume 156.
- Zimmerman, E. (2007). Creating a meaning-machine: The deck of stories called Life in the Garden. In Harrigan, P. and Wardrip-Fruin, N., editors, *Second Person: Role-Playing and Story in Games and Playable Media*. The MIT Press, Cambridge, Mass.