UNIVERSITY OF CALIFORNIA SANTA CRUZ

ALL THE WORLD'S A STAGE: A PLAYABLE MODEL OF SOCIAL INTERACTION INSPIRED BY DRAMATURGICAL ANALYSIS

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

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE

by

Joshua Allen McCoy

June 2012

The Dissertation of Joshua Allen McCoy
is approved:

Professor Michael Mateas, Chair

Professor Noah Wardrip-Fruin

Professor Stacy Marsella

Richard Evans

Tyrus Miller
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

All the World's a Stage: A Playable Model of Social Interaction Inspired by Dramaturgical Analysis

By Joshua Allen McCoy

The design of video games has been driven primarily by innovations in graphics technology and our ability to model physical spaces. However, the exploration of other areas of game design, such as interactive and social storyworlds, has been relatively slow. Games featuring complex story-driven gameplay are rare because we lack playable models of these design areas. Unlike graphics and physics simulations with their common language of mathematics, models of social interaction and storytelling have no such representation to serve as a basis for creating playable models of their domains. Additionally, we have little knowledge of how to present these types of gameplay to players as examples from which insight can be drawn are rare.

Current games that contain playable social interactions or storyworlds either have limited content caused by the large growth of story spaces to be authored, such as computer role playing games that use dialogue trees, or have abstract interactions that tell no concrete story, such as many social life simulation games. These games either offer a limited complexity of choices to the player with detailed realization of character

interactions in the storyworld or a significant amount of player choice complexity with abstract character interactions.

This problems are addressed in this dissertation via Comme il Faut (CiF), a playable model of social interaction and authoring approach for creating social storyworlds. Through drawing inspiration from social science, particularly dramaturgical analysis, CiF provides a knowledge represent that is able to encode and reason about social interaction in a way that is amenable to authoring storyworlds. This knowledge representation enables the retargeting of performances consisting of fully-realized character dialogue to contextually appropriate social situations. This allows for the creation of storyworlds with a new level of player choice complexity with detailed realization of character interactions. Additionally, the combination of a procedural playable model with performance retargeting provides a solution to problem of authoring storyworlds with high player choice complexity.

To explore the newly-opened area of design while testing and improving CiF, Prom Week, a critically acclaimed experimental game featuring social game play via CiF, was created and released to the public. Because the game features a large amount of content used to enable a large space of stories for the player to explore, this dissertation features descriptions of the design, authoring approach, and design insights learned while creating Prom Week. Additionally, this dissertation includes evaluations of Prom Week based on player traces to determine the both how Prom Week provides customized gameplay experience to players and to determine if players are exhibiting strategic play. This is followed by a comparison of authoring and complexity of story space of Prom Week with the best of the story-driven computer role playing games.

To compliment CiF's turn-based interactions, this dissertation features a description of the Holodeck system, a test bed for socially aware AI avatars taking the roles of supporting characters, and an evaluation explore the complications that arise when using social artificial intelligence systems in real-time environments.

Acknowledgements

This dissertation and line of research would not have been possible without the guidance and support of my advisor, Michael Mateas. Not only did he work hard to provide funding for graduate career, he was kind enough to let me embark on this journey that started with vague ideas about character social interaction and ended with an AI system, a great game, and a dissertation.

I consider Noah Wardrip-Fruin my second advisor as he was very much involved in *Prom Week* and was a crucial member of my dissertation committee. Noah consistently provided excellent advice and feedback throughout the last few years.

I would like to thank my Stacy Marsella and Richard Evans for both rounding out my committee in very valuable ways and for creating work that has inspired me for years.

The development team, or "tiger team," of *Prom Week* was talented group of individuals and the game would have been impossible to create without them. I would like to thank the core team of Mike Treanor, Ben Samuel, and Aaron Reed for their heroic efforts. The following undergraduate research assistants did incredibly hard work and are immensely talented: Ryan Andonian, Kathleen Kralowec, Corey DiMiceli, Zane Mariano, Jacob Pernell, Christian Ress, Duncan Bowsman, Travis Brown, Melissa Bernetsky, Devon Wyland, Ellen

Otsuka, Garin Kessler, Alexander Baker, Daniel Cetina, Alexander Schneider, and Lauren Scott.

Getting to UC Santa Cruz for graduate school in the first place was a tricky endeavor that required many resources. I would like to thank my parents, Steve and Melissa, for everything they have done for me over the course of my life. Melissa taught meunderstanding and love and Steve instilled a sense of responsibility and honor in my life.

Finally, I would like to thank Ariel for her unconditional love, caring, and incredible editing. She was the emotional support that has kept me afloat in my graduate studies.

Chapter 1

Introduction

1 The Design Space of Interactive Experiences

Innovation is important to many disciplines. In economics, innovation is held as critical to growth and the proper functioning of economic systems (Krugman 2007). Innovation leads fields like literature out of an easy blissful state and into one of challenge and fulfillment (League 2012). In interactive experiences and video games, there has been a perceived lack of innovation that is resulting in derivative work. In the face of graphical improvements and aesthetic experimentation, gameplay has been identified as the key area that is lacking in innovation (Blow 2006).

A more nuanced discussion on innovation in video games can be found in the Photoshop of AI debate (Hecker et al. 2009) and the ideas of expressive intelligence (Mateas 2003a) and expressive processing (Wardrip-Fruin 2009). Innovation is currently stunted as the community of interactive experience designers is stuck in a local minimum of design space. This minimum is characterized by our fluency of designing in Euclidean space (Mateas 2011). There are very effective and mature methods and tools for designing physical environments and simulating Newtonian physics for playable experiences. From high-fidelity simulations of physics to 3d modeling and animation tool chains, we have the scientific background knowledge and tools available to create highly playable and robust artifacts in this space.

The barriers around this minimum can be seen as the lack of knowledge of other spaces of play. One barrier can be seen in the current generation of storytelling games. These games have constrained story spaces but highly dynamic and playable combat and physical spaces. The difficulties of making computer role playing games (CRPGs) with gameplay intrinsically about the narrative and story characterizes the barrier; a barrier we can breach through making those spaces more playable distinctly. Designers' capabilities to deeply model stories and relationships in playable ways are nowhere near their capabilities to model experiences intrinsically about physical spaces.

Beyond these local minima in design space are areas that we do not know how to make playable or cannot envision. As seems to be the case with CRPGs, incremental progress in deep storytelling seems to be very slow; dialogue trees and cut scenes have been the standard for many years. This particular barrier may be emblematic of the other barriers keeping us in this minimum: they are too steep for incremental progress to overcome. Continuing the example of CRPGs, developers and players know that they want more playable stories[1]. Furthermore, scholars and authors have a long history of both studying narrative and creating deep, non-interactive media. However, interactive experience designers do not have the conceptual (let alone software) tools to bring the existing body of knowledge in the area to bear.

To overcome these types of steep barriers, the knowledge, systems, and tools must exist to make a large enough leap in the space of design to overtake the barriers of the local minimum. The creation of designable systems and authoring tools aimed at creating artifacts from the other side of the barrier need to be conceptualized and built. This is not an easy problem; physicists have spent a great deal of time and effort in determining the fundamental aspects of Euclidean space. Computer scientists have painstakingly modeled simulations and crafted tools for digitally manipulating virtual realms of physics. And, finally, game designers have dedicated years to making interesting, fun, and efficient models of physics with the appropriate authoring tools for interactive experiences. Similar effort needs to be spent to surmount the design barriers that prevent exploration.

---

[1]There are active and vibrant communities of CRPG players who use these games as a platform for role playing. Naturally, they want the games they play to be more conducive to their role playing needs:

http://j-u-i-c-e.hubpages.com/hub/roleplaying-skyrim
http://j-u-i-c-e.hubpages.com/hub/good-rpg
http://community.wizards.com/the_jester/blog/2010/06/26/nitpicking_the_roles

One of the attributes that makes Euclidean spaces playable is that they are easy to model with mathematics based on continuous numeric values (which could be largely responsible for vector fetishism found in video game programming) which happens to be, or possibly was inherited from, the emphasis on statistical and scientific computation that largely overlapped with the development of playable physical spaces in video games[2]. While some academic communities were figuring out how to build Beowulf clusters, determining the top 500 list of supercomputers, and using that hardware to solve complicated numerically-based problems, video game designers spent significant time and resources modeling physics in 3d space while borrowing techniques used in scientific computing. The result is clearly seen in what spaces are playable in video games. Modeling numeric systems of combat and physics is common place while there is a lack of models of narrative and story spaces. Stories and narrative seem to be hard to represent with float values.

The tools designers and developers use to create games are mature and use the common vernaculars of physical relations and graphical rendering techniques. With exemplars like Maya and Photoshop, comprehensive and competent tools have been created and established as standards in their domains. Additionally, these tools have the

[2] http://grandtextauto.org/2004/11/10/the-numerist-fallacy/

ability to be modified with external plug-in functionality which gives their users the capability to construct task or domain-specific functionality. These plug-ins can be shared with other users which serves to extend the capability of these programs. The common language of a mathematical base provides the context needed to make plug-ins usable across projects.

In the space of interactive stories, there is no common language usable across tools. There exists no neat and tidy mathematical underpinning (as yet discovered) to make the parts of a whole story mesh together seamlessly. The problems intrinsic to interactive stories are not the same as those used in scientific computation. Quick calculations of floating point numbers are less important than taking contextual and relational considerations into account. The important fact is not how fast a well-shaded plasma bullet travels along a trajectory to its target; the personal, social, and narrative-based reasons why a character would choose to take up a plasma rifle and commit violence are very important to storytelling. Not only are there no tools to capture these types of story considerations, there is no accepted body of knowledge to make a common vernacular on which to construct the story equivalent of a graphics or physics engine. Past this establishment of a vernacular, the cultural rules of story worlds can differ dramatically and provide a shifting foundation that would make the creation of story plug-ins difficult.

Because of this dominant culture of computation during key times in the formation of video games as an industry and robust medium, moving forward in many directions of the design space also means transforming or breaking free of the norms associated with those formative cultural forces. As such, acknowledging the current state of design, including its strengths and weaknesses, provides critical perspective from which future design areas can be seen. The focus of this dissertation is on the technological research in and applications of artificial intelligence systems that challenge these cultural predispositions while breaking through a design barrier to open up new possibilities in the design space of video games and interactive experiences.

The research in this dissertation addresses these issues that prevent innovation in the design space of interactive experience by providing a system, authoring approach, and implemented game that open up a large design area that has yet to be explored. This work avoids the problems and design barriers associated with incremental design space exploration as it takes a large leap into unexplored design areas via making a new realm of play possible. This is made possible by making systems, tools, and authoring approaches that fill in the gaps needed to create a playable system for interactive storytelling. Graphics and physics engines of modern video games rely on understanding the subject through centuries of scientific inquiry; a common language of mathematics to use that knowledge in comprehensive ways; computational models that can simulate the mathematical understanding; and design applied to the simulations to create engines for games. In interactive storytelling, there exists only the first part of this process, understanding, with the work done in social science. This dissertation fills in the gaps of a common language via ethnomethodology and knowledge representation (which comprise the foundation for an AI system as well as providing an approach to authoring in this space); enables simulation via AI techniques; and applies design to the simulation to create an interactive experience.

2 Enter Interaction

Video games and Interactive Drama (ID), as with books, radio, and movies, are media in which compelling stories can be told. However, storytelling in virtual worlds has a degree of complexity that the previous forms of "static" media do not encounter: audience interactivity. The ideological shift from audience member to an actively participating player has far reaching impacts on the resultant experience and the design of creating an experience in video games or IDs. The ability of the player to interact meaningfully and have that interaction change the virtual world in impactful and interesting ways (Sims 1999) can allow for both exploration of the world and a higher degree of connection between the player and the unfolding story (Murray 1997).

Figure 1 – The "finger cut" scene from *Heavy Rain*. The player is asked to cut a finger off of the character they control. Interaction enhances the player's experience by making the entire scene playable through abstract control (such as the "TRY IT" and *"DO IT"* commands).

Final Fantasy VII makes the relationship of the player with Aeris more potent by making her playable. By putting the player in control of the actions of Aeris (even if at only the level of map navigation, gear/skill selection, and combat), a potent type of relationship is formed. Many players experienced a deep emotional impact through this type of interactive storytelling. Another example, *Heavy Rain* (Quantic Dream 2010) makes actions ranging from highly dramatic to mundane playable through abstract control schema. This was done intentionally to bring the player into the story through the interaction (Nutt 2012). This game includes a well-known scene where the player is asked to cut off the finger of the character they control to save that character's son. The player has abstract control over choosing a cutting implement, lowering the implement to the finger, and “DO IT” or “TRY IT” commands to attempt the deed (Figure 1 is a screenshot of this scene).

The benefits of including interaction in storytelling come with a price. For an interaction to be meaningful, a decision needs to be made by the player that has some impact on the virtual world; if the player had no impact on the story world, it would not truly be an interactive experience. This would be indicative of an interactive loop with an unhealthy flow of information between the player and the system (Crawford 2002). With each authored point of player interaction, different potential stories are created through the resulting impact of the player’s choice. As the player could have selected from any of the choices available at each decision point, each individual choice must be accounted for and responded to properly by the simulation of the virtual story world. In effect, authoring an ID is authoring a space of possible stories and playing an ID is exploring one distinct story of the many possible that could be experienced.

To make the space of stories encompassed by the ID as compelling as possible, the player should have a number of interaction options available that are consistent with the state of the story world. As each decision has impact on the story, each time the player interacts with the world, the effects of past player choices need to be accounted for. Accounting for past impacts on the story as the story unfolds appears to leads to an exponential explosion in required authoring. At worst every possible choice of interaction the player is presented with has to distinctly account for every possible action that could have previous taken place. If each one of these possible states of player consequence is to be explicitly detailed by an author, the burden on that author would quickly become intractable for even small storyworlds. This authorial burden is a key problem facing the future of IDs as a form of art and storytelling.

A parallel problem to authorial burden inherent in creating a story space is that of narrative coherence (Trabasso and van Den Broek 1985) of the story world. With the complexity of authoring spaces of stories, it would be very easy for distinct world states or subspaces of the space to be overlooked in some way by the author; plot inconsistencies, breaking of the social norms of the world, or interaction options that may not be physically possible in the story world are all likely breaks in consistency. Additionally, the believability and richness of characters in the story world play a key role in narrative coherence. Keeping the actions of the character within the bounds of normative social behavior (Si and Marsella 2010) is of similar importance to the audience as plot consistency. Preventing, controlling, and correcting these lapses in story world consistency is a key factor in properly managing the authorial burden required to create IDs.

This dissertation presents both an authoring strategy for and an AI system implementation of a playable social model and how it procedurally manages the social state of story worlds while addressing the problems of moderating character behavior with

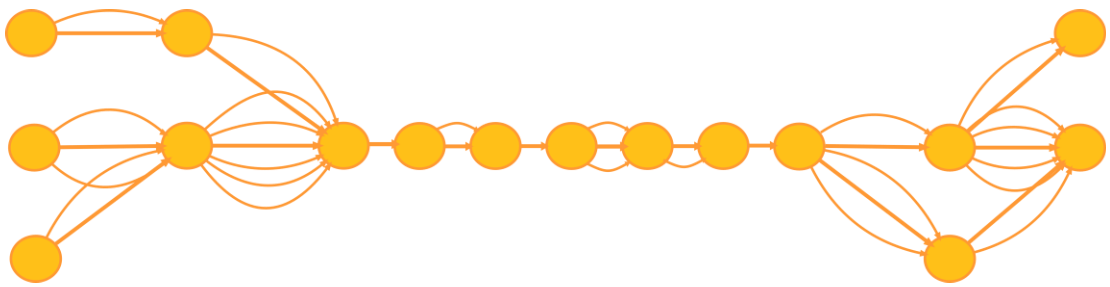


Figure 2 – A visualization of the "beads-on-a-string" method of narrative structure found commonly in computer role playing games. Notice the distinct starting points on the left that sync in to common story nodes in the middle, the large number of nodes in the middle that do not branch, and the end branches to a small number of terminal nodes.

respect to a view of social norms. To address the problems of controlling the quality and coherence when authoring large, complex stories and the associated super linear increases in authoring burden, my work pushes forward the state of the art of interactive storytelling by drastically increasing the complexity of choice available to players in the social aspects of the storyworld.

3 A Playable Model of Social Physics

It is likely that inconsistencies in the social milieu will be commonly created in procedurally generated stories. The consequences of such inconsistencies are a break with what the player expects or will accept from actions taken by characters in an interactive narrative. Characters breaking the patterns of normal social behavior have a high chance of harming the player's agency and enjoyment of their play experience.

As social contact with other entities is a core feature of IDs, preventing and resolving social inconsistencies during authoring or performance is very important. When authoring story spaces, a social space is constructed; social norms, cultural spaces, character relationships, culturally significant objects, and societal expectations are all fundamental aspects of storyworlds. Through establishing a story world with its social space, the author implicitly defines a contract with the player in which the player suspends their disbelief in exchange for the characters behaving in ways consistent with the story world. In effect, the contract requires the author to maintain each of the social and story contracts between the cultural space and the players and how they perform.

To handle the authoring and social complexity of large stories, most interactive narratives rely on the "beads-on-a-string" approach (as seen in Figure 2) where sections of relatively free player action are followed by static cut scenes that rarely account for the action the player previously took (Harrigan and Wardrip-Fruin 2007). To introduce some variation, some games give players a choice between several explicitly authored paths, but, as discussed in the previous section, this approach quickly results in an impossibly large space to author for that is exponential to the number of interactions accounted for. To minimize this problem of authorial burden and to make play experiences more varied, computational models, such as simulations of the physical world, can be used to provide a space of interaction and resulting stories. Computational models make authoring more tractable by allowing the system to manage the mechanics of the space, thus reducing the number of interaction combinations for which one must explicitly author.

Stories often revolve around the relationships of the characters in the cast. The relationship of conflict between the protagonist and antagonist, the romance between main characters, or the differences between a character and society are all common relationships in strong narratives. As the field of sociology has developed rich and varied models of human social interaction, a way to enrich the story space and decrease the work required to author an ID would be to make social interaction an explicit part of the virtual world and authoring process. For this to be effective, a playable model of social interaction must be developed.

As discussed earlier, video games have achieved a high level of playability in physical spaces; combat, movement, and physics-based puzzles are all very playable and well modeled. Imagine a world in which social spaces are playable to the same degree as physical spaces currently are. A new and powerful set of playable media experiences could establish new genres of gaming and interactive art. The goal would not be to recreate the everyday social world, but to create social dynamics specifically crafted for use in media — just as platforming games do not reproduce the physics of the everyday world (but rather tune physics for gameplay) and fiction writers portray behavior and dialogue in stylized fashions that differ markedly from the average conversation.

To minimize this problem of authorial burden and to make social aspects of interactive stories more varied, we need computational models of social space and social interaction that are as complex as their physical space computational model counterparts. This would make building social play experiences more tractable by allowing the system to manage the mechanics of the social space interactions, reducing the number of social interactions that need to be explicitly authored. With most virtual storyworlds, the social space is implicitly encoded in the affordances presented to the player and the responses from the characters. Making the socially modeled content explicitly defined requires some mode of representation coherent enough to take what is known about human social behavior (such as in the field of sociology) and strong enough to be useful computationally.

However, representing social interactions, cultural context, and the consequential interactions between the two in their entirety is an impractical, or perhaps impossible, task. This task of representation is similar to that of representing the possible combinations of player interactions and their impacts in the virtual world. The space of contexts (states of the virtual world) and social interactions (player interactions) is prohibitively large — and not necessarily the ideal one for authorial expression. As a result, knowledge representations and architectures rich enough to support plausible social interactions, that are tractable to implement, and that support authored variation are needed.

In the initial work toward creating a playable social model, I leveraged Goffman's dramaturgical analysis[3] (Goffman 1959) to encode patterns of normal social behavior to create a useful abstraction for enabling social play—we call these social exchanges (McCoy and Mateas 2009). Growing from their definition coined during the development of *Façade* (Mateas and Stern 2005), social exchanges have been defined as multi-character social interactions whose function is to modify the social state existing within and across the participants. Social exchanges are designed to encode normal patterns of social behavior while providing space for personality-specific character behavior in a format that an AI system can make use of. Additionally, social exchanges are intended to be an abstraction over the implicitly encoded interactions found in *Façade* and allow for an explicit encoding that can be used by many characters in the virtual world without breaking the audience's expectations (social norms) for characters' performances.

4 Research Contributions

The first contribution of this work is:

[3] The choice of using Goffman's work as a conceptual foundation for the playable model of social interaction featured in this dissertation was based on the stories we wanted to tell; highly dramatic, detailed stories about a small group of characters. Additionally, the medium for this work was interactive video games complete with virtual worlds. Dramaturgical analysis is at the correct scale (contextually appropriate small group social interactions) and has conceptual room for a virtual world and its contents by including concepts such as props and stages.

The development of an authoring approach for creating domain models for making social interactions playable and the demonstration of this approach being used to model fictional domains and virtual world domains.

This dissertation features the implementation and design the domain models resulted that in *Comme il Faut* (*CiF)*, an AI system implementation of a playable model of social interaction, and *Prom Week*, a video game whose gameplay revolves around *CiF*'s capabilities. To compliment *CiF's* turn-based interactions, the *Holodeck* system, a test bed for socially aware AI avatars taking the roles of supporting characters, and its evaluation explore the complications that arise when using social AI systems in real-time environments. Lastly, the dissertation contains examples and analysis of the authoring processes for *Prom Week* in *CiF* as well as using visual analysis of human role players in creating the AI agents in *Holodeck.*

The second contribution of this work is:

Authoring was made possible for both the space of playable social interactions and for a new level of player choice complexity. In addition, a tool and authoring methods were developed that allowed previously untrained authors to create complex content.

With the previously mentioned problems with standard CRPG tools, I believe it is not possible to create the type of complexity enabled by *CiF*. Even with the amount of authoring resources brought to bear on video games like *Star Wars: The Old Republic* (BioWare 2011), the complexity of player choice is less than those in *Prom Week*. This work reflects my belief that a new level of playable story spaces is desired by players and is not being addressed by current CRPGs. *CiF* was designed to address this problem by giving players dramatically more choices and opportunities to interact with the social aspects of storyworlds. *Prom Week* was developed to serve as "proof by implementation" of *CiF*'s effectiveness.

In this dissertation, I contribute to both research and design. The design lessons learned through building these systems provide insight to future forays into building games with playable models of social interaction. Before we built *Prom Week* and its prototypes, there were design issues that could not be foreseen: the challenge would it present to players; the difficulties in constructing an initial social state; and the difficulty in conveying backstory are a few of the many examples.

5 Document Structure

In chapter 2, Related Work, the contributions and systems described in this dissertation are situated in relation following related areas: playable models of social interaction in AI research; playable models of social interaction in video games and interactive experiences; social sciences; and the humanities.

Chapter 3, *Comme il Faut*, describes *CiF*, the AI system implementation of a playable social model. The design goals, a general overview of functionality, a detailed description of the knowledge representation, and a thorough explanation of the internal processes of *CiF* are presented. This chapter describes work done with Mike Traenor, Ben Samuel, Aaron Reed, Noah Wardrip-Fruin, and Michael Mateas (McCoy and Mateas 2008; McCoy, Mateas, and Wardrip-fruin 2009; McCoy, Treanor, Samuel, Tearse, Mateas, and Wardrip-Fruin 2010; McCoy, Treanor, Samuel, Tearse, Mateas, and Wardrip-fruin 2010).

Chapter 4, *Prom Week*, is a description of *Prom Week,* the video game built around the affordances granted by *CiF*. This chapter starts with explaining the design goals and motivations behind the development of *Prom Week*. Next, the basic gameplay is described and is followed by an overview of the game's fictional world. Following is a description with examples of how content for *Prom Week* was authored with *CiF.* This is followed by how *Prom Week* uses *CiF* outside of its simulated process to enhance gameplay. The development of *Prom Week* was a team effort. The primary development team consisted of Mike Treanor, Ben Samuel, Aaron Reed, Noah Wardrip-Fruin, and Michael Mateas (McCoy et al. 2012).

Chapter 5, Evaluation of *Prom Week*, is a critical examination of how *Prom Week* and *CiF* address the previously stated research questions. The first research contribution is evaluated through an analysis of the game traces gathered from players playing the publicly released version of *Prom Week*, which was conducted with the following collaborators: Ben Samuel, Serdar Sali, Noah Wardrip-Fruin, and Michael Mateas*.* The second contribution is evaluated by a comparison betwee the player choice complexity of current story-driven CRPGs and the complexity of *Prom Week*.

Chapter 6, *Holodeck*, describes the *Holodeck* project and its evaluation. Aimed at exploring supporting character realism (SCR), this project involved the design and implementation of AI-driven avatars in *Second Life* (Lindon Lab Inc. 2003). This work was done in conjunction with James Skorupski, Michael Mateas, Catherine Zanbaka, Kathy Ryall, and Byran Loyall (Skorupski and McCoy 2012).

Chapter 7, Authoring Content for Playable Social Models, starts with a general method for authoring fictional works with *CiF*. The rest of the chapter is dedicated to case examples for authoring playable social models including preliminary analysis of dramatic media that formed a foundation of knowledge representation for *CiF*, the avatars in the *Holodeck* project, and using *CiF* to author *Prom Week*'s fictional world.

Finally, in chapter 8, Conclusion and Future Work, the contributions of this dissertation are summarized and future directions of this line of research are presented.

Chapter 2

Related Work

Like other design endeavors, the act of designing games naturally engenders the desire to explore that design space. Often these impulses have taken forms of inquiry that resemble an informal version of biological taxonomy; categorizing specimens based on their common attributes. This way of understanding a space is natural to people in much the same way stereotyping other humans is natural. Looking for quickly recognizable patterns in what is experienced in everyday life helps a person in expeditiously evaluating and responding to new encounters. If a person knows what a shoot-em-up (shmup) looks like and knows what affordances are typical of that type of game, shmups seen in the future will be easier to acclimate to. However, stereotyping has the weakness of allowing for misclassifications. As a taxonomy can ignore important details while simultaneously placing undue expectations on the classified, the categorization of games into genres has similar drawbacks in its stereotyping process.

The establishment of genres presents its own barrier to game design innovation. The addition of new genres to the zeitgeist of design happens retroactively after the genre has been explored. For example, the recognition of the rhythm game genre was widely accepted in the west after games like *PaRappa the Rapper* and *Beatmania* explored the genre.

Well after the development of many games in the genre, rhythm games was not commonly included in descriptions of major video game genres (Rollings and Adams 2003; Perry and DeMaria 2009; III and Ogden 2005). As useful as genres are in understanding the existing landscape of design, they can put undo connotations on the reception of new games with innovative designs; the lag in the acknowledgement of new genres results in old genre labels being placed on games that are really the first members of a new genre.

Formal Abstract Design Tools (FADT) (Church 1999) takes a step away from the Bricoleur's (Lévi-Strauss 1966) way of understanding game design and moves toward a more rigorous study of what games are made of. Backed by the notion that "a common design vocabulary is the primary obstacle to having innovation in game design", FADT calls for conscious thought over what affordances you have as a game designer and how those affordances impact the experience of the player.

Supporting gameplay that opens up new spaces of game design via software simulation is an important step toward understanding spaces of design; where FADT looks at how existing design affordances impact the player's experience, simulations force a detailed exploration of the modeled affordance. Through the formalization inherent in modeling a particular domain in code, the details must be considered and represented. As the simulation approximations approach the fidelity of the actual space, the smallest details are consciously formalized and put in relation to all of the other considerations in the model. Making more design spaces playable through the virtual modeling of their domains is a way to overcome design barriers by way of increasing our understanding of those areas.

As we have some spaces, such as physics, understood conceptually and satisfactorily modeled in software, this successful paradigm of creating compelling and playable models should be applied to exploring the space of game design. To achieve this deep level of interaction with story and social spaces involving characters, these playable models need to represent the story world and its characters in a way that preserves the contract players and audiences expect from other forms of storytelling media. Reasonable motivations for the actions of characters in the story world that transform the plot in compelling and understandable ways are necessary to tell interesting stories. According to Crawford, stories that are set up properly will commonly snap into place in the player's mind 90% of the way through the experience (Crawford 2004). This type of reasoning is what AI systems and techniques are designed to perform and is what imbues a character with a perceivable purpose in the mind of the player.

Systems that are completely predictable to the player feel mechanical and "dead" (Mateas 2003b). For instance, an object that has been dropped will act appropriately for the physics of the environment it is in, but it will always drop the same way and it will always bounce the same way when it hits a surface. The player does not assign meaning to the actions of the ball, does not need to try to understand the psychological goals of the ball, but instead understands that there is a physical system that is operating on the ball. While modeling this behavior is important for believability of the world, it does not add life to the system.

Contrast this with human behavior, which is so complex that it cannot be fully predicted. We can rationalize and reason about why someone acted the way they did, but their exact actions cannot be forseen. While there is some debate about whether humans truly have free will, or if they are ruled by chemical reactions in the brain (Nutt 2011), the human mind is so complex that it can be neither fully modeled nor predicted. Dennett (Dennet 1987) argues that when interacting with a complex system, humans will abstract away the complexity and instead reason about the intentions and goals of that system.

Likewise, when players encounter an AI system in a game, they assign intentionality to that system, "using words whose meanings go beyond the mathematical structures" (Agre 1997). They create narratives that rationalize the AI's actions and reasoning about the AI's goals (Sengers 2000). This intentionality can be observed both through the actions of an explicit AI-controlled character in a game, such as an enemy NPC in *Halo 3* (Bungie 2008), or through the results of actions taken by an invisible agent that responds to a player or designs part of the world, such as the level generator in *Rogue* (Toy et al. 1980) or the drawing analyzer in *Crayon Physics Deluxe* (Purho 2009).

When an AI system does not have sufficient complexity to support the intentionality the player reads into it, the illusion (the breakdown of which is the *Eliza effect*, extensively described by, among others, Wardrip-Fruin (Wardrip-Fruin 2009)) breaks down and the life-like impression that has been built up by the AI is lost. This happens because the player now understands the system enough that there is no longer a need for the player to abstract to an intentional level. Robust modeling and implementation of behavior on the other hand, be it believable social interaction or the creative design of a level, can lead to complex areas of exploration from which to draw new game experiences. I believe it is possible to find much new ground in the space of possible games by exploring AI systems.

1 Playable Models of Social Interaction

1.1 Games with Playable Social Interaction

Gameplay that gives the player affordances to make deep and impactful changes to the social landscape of the virtual world is rare to encounter and difficult to achieve. Partial social play is exhibited in some playable experiences. In computer role playing games (CRPGs), such as *Dragon Age: Origins* (Bioware 2009), the social interaction is largely implicit in the dialogue trees and combat system. As social play is at best present in some dialogue tree choices, CRPGs are in the position of providing well crafted, primarily linear social performances that gives the player very little room to interact. A reason why players have so little influence on the social world in these games is because the authoring required to provide more social space to play is too large. As mentioned in the introduction, dialogue trees are the predominant tool used in crafting the socially interactive parts of these story worlds. While this is a convenient method for encoding a certain size of dialogue with player interaction, each additional degree of complexity requires more exponentially authoring than the previous degree.

In *Spy Party,* an "espionage game about subtle behavior, deception, performance, and perception" (Hecker 2010), the gameplay hinges on either blending in with the normal social behavior found in a cocktail party while accomplishing a goal as a spy or being a sniper that picks out breaches of those norms to uncover the spy. While there is a playable space where the player either interprets or performs social roles, there is no system actively modeling social interaction. The social conveyance is a production of NPCs with individually crafted behaviors and human controlled characters.

The IGF Nuovo winning game *Storyteller* (Benmergui 2012) features a limited playable story model where the player creates a story based on a cue or a story goal. Given goals like "serial heartbreaker" or "black widow" and a set of comic book-style panes, the player uses story objects like characters, props, and settings to create a story that matches the goal. *Storyteller* reasons over player-created story via an interdependent set of rules (death is dependency for determining grief for example). In comparison to *CiF*, the heart of the

Storyteller's AI system is a small number of interdependent rules that capture the story consequences of player input and are used to post-process the game state. As the world and characters were designed to be as generic as possible, the characters have no agency and the set of dependency rules are solely responsible for the system inferring story from player-created game state.

The Sims 3 is an example of a culturally influential and commercially successful video game that has a highly dynamic social space (The Sims Studio 2009). Its characters, known as Sims, have traits and desires that inform the social practices (social norms and clusters of expectations) they perform (Evans 2008). These practices have both a regulating and constitutive effect on character behavior: they help specify how an agent should not behave as well as enable new behavior and intentions. A Sim can be involved in more than one practice at a time and a practice can involve more than one Sim at a time. These practices are first sketched in Unified Markup Language (UML), then are encoded through a design language that is similar in function to propositional logic. The social practices provide context for design statements and have a similar relationship to social exchanges and their constituent events. Two major differences between *The Sims 3* and *CiF* is in the complexity of the statements of social norms and the use of history in those statements. *CiF* provides a level of complexity similar to first order logic in that parties outside of the practice can be referenced (**x** is cheating on **y** if **x** and **y** are dating and there is a character **z** also dating x). *CiF* allows for both back story (history of the story world before the player is involved) and play history to be used in reasoning and social exchange play. This also allows for changes in the social state to result in cascading social consequences. As seen in chapter 3's Design Goals section, *CiF* enables concrete rather than abstract character performance.

Storybricks is an in-development system that embeds psychological effects and attributes into not only the characters of the world but in to any world entity or abstraction that has the capability to influence the behaviors of characters. Based on the five-factor model (McCrae and John 1992), objects are associated with vectors of numbers that represent the influence those objects have on the psychological states of entities in the world. For example, a tavern could be associated with a positive modification to the extrovert attitude (Bura 2012). The effect of this psychological modification would be that the tavern is a place of more frequent character interactions as every character in the tavern becomes more extroverted. Additionally, *Storybricks* is unique in that it contains a visual behavior authoring tool that can be used in real time. In comparison to *Storybricks*, *CiF-RPG*[4] (Sullivan et al. 2011) allows for the embedding of cultural knowledge and social reasoning in objects in the storyworld. As discussed in the Social Science Foundation of *CiF*

[4] During the authoring of this dissertation, *CiF* and *CiF-RPG* are being functionally merged to make a singular, canonical version.

section later in this chapter, *CiF* was designed as a response to the downsides of using vectors of real numbers to bring playability to storyworlds. These downsides include the reduction of the represented space into binary decisions (like if the extraversion value is above 4.0, behave gregariously) and wildly oscillating behaviors caused by small changes in the float values of vectors (like the extroversion value rapidly bouncing between the range of 3.9 and 4.1).

Perfect Circle: The Quest for the Rainbow Pearl[5] is a game where the player controls a group of characters (each of which have distinct capabilities and skills) with the goal of searching the world for a magic item (Prada and Paiva 2005a). If each character acts independently, the magic item may not be found. The characters can perform socio-emotional group discussion to coordinate their actions. This group interaction is simulated via the synthetic group dynamics model (*SGD)* of multi-agent social interaction (Prada and Paiva 2005b; Prada and Paiva 2005a). This model is based on having each group member being aware of the other group members and of the group itself (Prada and Paiva 2009). Each character has a model of the group consisting of four levels: the individual (the individual characteristics of agents), group (underlying structure of the group), interactions (classes of interactions and their dynamics), and context levels (nature and environment of the tasks to be done).

[5] http://gaips.inesc-id.pt/~rprada/perfect-circle/

The individual level consists of a character's skills to complete the game's tasks and a personality description based on the five-factor model (McCrae and John 1992). The group level uses social influence and attraction to determine the agents position, or strength of the character's actions, in the group. The interaction level is based on Bales Interaction Process Analysis (Bales 1951), theories of social power (French and Raven 1968), and Heider's balance theory (Heider 1982). Bale's work is used to categorize the group functions of interactions as agree, encourage, disagree, and discourage. Facilitate problem, obstruct problem, gain competence, and loose competence are instrumental interactions added to support character skills. The dynamics of social interactions encode declarative statements of character motivations for the results of interactions[6]. This encoding of interaction dynamics is comparable to *CiF*'s influence rules and triggers. Unlike *CiF*, *SGD* does not reason over history. Additionally, the interactions do not have an abstraction equivalent of social exchanges. In *Perfect Circle*, the actions are referred to abstractly ("proposal and discussion finished") while *Prom Week* features fully-realized dialogue and animated performances.

---

[6] One interaction rule in *SGD* represents that agent extraversion, group position, and motivation affect how likely that agent is to start an interaction in the group. The following is a formalization of that rule (Prada and Paiva 2005b):

$$\forall Group(G) \wedge Interaction(I) \ \wedge \forall Members(G) : \\ Extravert(A) \wedge GroupPosition(A,G) \wedge \\ Motivation(A,G) \vdash Start(A,I,G)$$

Social exchanges and *CiF* exist to enable playable models of social life informed by social norms. Other playable models exist that include play over social interactions between agents. The interactive narrative that is closest to offering a high level of social play is the experimental game and ID, *Façade* (Mateas and Stern 2005). The gameplay in *Façade* revolves around social play between the characters and the player. The patterns of social action and social norms of the characters in *Façade* are implicitly encoded in behaviors written in *A Behavior Language* (*ABL*) (Mateas and Stern 2002). *Façade* is a game of limited duration that has two well-defined social exchanges (an "affinity game" and "therapy game"). These social exchanges are constructed with dramatic beats by the human author specifying content selection. In *CiF*, the system does the content selection using more general and abstract models of social exchanges.

Even with a carefully limited scope, the authoring process for creating *Façade* was very time intensive and would not scale well to a larger work of fiction. Further progress requires an approach that can address the authoring bottlenecks found in all approaches, while defining a path toward the deeper social gameplay found in *Façade*. The research around social exchanges and *CiF* is aimed at addressing these issues raised by *Façade* (McCoy, Mateas, and Wardrip-fruin 2009).

Cotillion is an in-development game that features textual and procedural storytelling in a Regency era setting (Alexander 2012). The player controls how a group of characters behave while the system responds to the player's choices by generating the next section of story. Based on Sack's work on conversations (Sacks 1974), the AI system in *Cotillion* generates a story while simultaneously keeping the interactions of the characters within the social expectations of the setting. As a contemporary to *Prom Week*, the two games have much in common at the system level – *Prom Week* focuses more on a simulation and puzzle solving gameplay, while telling coherent, believable, and well-written stories are at the core of *Cotillion*'s design. The two games share a point-based resource system that allows the player to modify the actions of the characters. Key differences between the two games are the presentation of storyworld to players and the possible scope of a gameplay session. The presentation of *Cotillion* is text-based and reminiscent of interactive fiction where as *Prom Week* has animated 2d representation of its characters that perform social interactions. In *Cotillion*, a session is the story equivalent of a scene in a chapter, such as a dining room conversation scene. As it is meant to be a simulation in which histories are built, what *Prom Week* players experience is closer to a chapter or a short story.

EL, a deontic exclusion logic used by *Cotillion*, is meant to address the inadequacies of hierarchical finite state machines when working with large scale multi-agent systems. A major strength of *EL* is that it declaratively represents why transitions in behavior take place. As it is deontic, *EL* "distinguishes between what is in fact the case and what should be the case" (Evans 2011). The following is an example of a conditional judgment in *EL* stating that x should eat y if it is edible and x has an empty stomach:

$$x: Stomach: Filled. Empty \wedge y: Edible \rightarrow [\,] x: Act. Eat. y$$

EL is a language that shares similarities to *CiF*'s influence and is meant to generate hierarchical finite state machines after being processed. While *CiF* relies upon weight and intent pairs on the right hand side of a rule to determine character desires, *EL*'s can infer what a character should be doing. Like influence rules, declarative conditionals can be added piecemeal so that new reasoning capabilities can be added without having to rewrite those that already exist. This allows authoring to progress in a linear fashion instead of the super-linear or exponential burden needed to add capabilities to finite state machines.

Created by the author of *EL, L* is a programming language meant to capture social norms through deontic and epistemic logics (Evans 2009). In response to problems with the score keeping, token/type approach, *L* is reification of an ethnomethodological approach to social practice; norms and public pressures both dictate what behaviors are appropriate as well as the revelation of new behaviors. Though *CiF* has yet no capacity for revelation, the behaviors of its characters are strongly derived from the social norms of the interaction of the character, history, and the story world.

The experimental CRPG *Mismanor* (Sullivan et al. 2011) is game built to explore the *Grail Framework* which includes *Grail Game Manager* (*GrailGM*) and *CiF-RPG*. This work includes brainstorming and game design tools that enable designers to author high-level rules from pieces of traditional content. *Mismanor* supports non-combat play, dynamic quest selection, and multiple re-orderable storylines, while telling a consistent higher level story. *CiF-RPG* is based on *CiF* with additional capabilities to fit CRPG conventions, such as game objects, items, and character inventory. Where *Prom Week* puts emphasis on strategic social play in a simulation of a high school storyworld, *Mismanor* uses *GrailGM* and *CiF-RPG* to tell stories and innovate on the nature of quests in CRPGs. Also, the games vary in structure; *Mismanor* is a single, crafted story and *Prom Week* contains a series of stories that have goals for specific characters.

1.2 Research Playable Social Models

Because they are conceptualized and constructed to achieve research goals, academic models of social interaction tend to dig deeper into modeling domains. However, they typically do not have an emphasis on producing focused and polished playable experiences with their implementations. Largely the focus is on evaluating the efficacy of a model with respect to the research question and evaluation methods that were delineated in the early stages of the research to which the model belongs. However, there are several exceptional projects that have resulted in both valuable research contributions and playable models of social interaction.

PsychSim is a multi-agent social simulation system that models social interaction with an emphasis on managing the conflicting goals that arise from complex, character-relative goal priorities in the face of uncertainty by employing decision theoretic methods (Marsella, Pynadath, and Read 2004). Through using recursive modeling of other agents, *PsychSim* employs an implementation of theory of mind to enable complex reasoning about other agents based on subjective knowledge (Pynadath and Marsella 2005; Marsella and Pynadath 2005). This system also has the ability to take an incomplete script consisting of a set of desired behaviors and tune the characters' goal weights to values that will accomplish the script when simulated (Pynadath and Marsella 2004).

A framework for designing interactive narratives based on *PsychSim* known as *Thespian* allows for authored plot development while simultaneously enabling the development of rich characters (Si, Marsella, and Pynadath 2005a; Si, Marsella, and Pynadath 2005b). *Thespian* has also been applied to the authorial burden problem (Si, Marsella, and Pynadath 2007), appraisal in theory of mind (Si, Marsella, and Pynadath 2009a) and directorial control over IDs (Si, Marsella, and Pynadath 2010a). As a social simulation, *Thespian* models aspects of socially normative behavior in the domains of obligation management, conversational turn taking, conversational flow, and character affinity (Si, Stacy Marsella, and Pynadath 2006) to better enable rich character and narrative coherence (Si and Marsella 2010). The *Thespian* framework is a playable model in that it has been used in a number of interactive narratives including the *Tactical Language Training System* (Johnson et al. 2004) and an adaptation of *Little Red Riding Hood* (Si, Marsella, and Pynadath 2010b). In comparison to *Thespian, CiF* differs in the way that characters determine their actions. *Thespian* characters use the per-agent policies of *PsychSim* to determine the next action to take that is a maximum of expected reward based on the relative value of characters' goals. *CiF* relies almost exclusively on the social norms declared outside of the character to inform the character of the actions that are best in keeping with those norms. Instead of characters being consistent with their internal goal and policy sets, *CiF* characters have consistency enforced at the normative behavior level.

The *Pataphysic Institute* is a playable model of psychology in the form of a massively multi-player online role-playing game (MMORPG) (Eladhari 2009) where the game mechanics are based on the personalities of the inhabitants . *Mind Module* allows for the gameplay of the *Pataphysics Institute* to be intricately intertwined with the mood, emotions, and psychological state of its cast of characters and the players. To situate the player in this environment of mental physics, the player is given a questionnaire to determine the mental capacities the player will have when interacting in the virtual world. The deep linking that *Pataphysics Institute* has between its mental physics and gameplay is the same goal we wish to use *CiF* to achieve in *Prom Week*.

Similar to *CiF* being a playable model of social physics, *Mind Module* is a semi-autonomous agent architecture that uses personality psychology and affect theory (Eladhari and Sellers 2008). This system was implemented as a per-character spreading activation network consisting of a weighted network of interconnected affect nodes. The four affect nodes types are personality traits, emotions, sentiments, and moods. Nodes are placed in the spreading activation network according to a character's personality and gameplay events. When affect nodes are activated, they affect nearby nodes where the impact decreases with distance. By having the affect nodes in a spreading activation network associated with a character via a personality test (if a player is controlling the character) and gameplay events, the spreading process provides results customized to each gameplay experience.

FearNot! (R. S. Aylett et al. 2005) is an application of virtual drama aimed at anti-bullying that has similar themes to *CiF's* architecture. Just as the concept of social exchanges is informed by drama via dramaturgical analysis, *FearNot!* is inspired by drama in the form of Forum Theatre[7] (Boal 1985). Driving this virtual drama is the AI system *FearNot!*

[7] Forum Theater modifies the traditional relationship between actors and audience. Breaking with the idea of passive audience members, the performance is based on the interaction of actors and "spect-actors", who can call for modifications to the actors' performances. All participants determine what the important issues are and develop themes. Role-playing serves as an analytic tool for debate and searching for solutions to problems raised by the themes. This process is an

Affective Mind Architecture (FAtiMa) (Lim et al. 2008; Dias and Paiva 2005). *FAtiMA* models psychological appraisal using a continuous planner (R. Aylett, Dias, and Paiva 2006) and OCC emotional influences (Ortony, Clore, and Collins 1990). When *FearNot!*'s interaction is viewed in terms of social exchanges, its internal structure is comprised of several social interactions: player introduction, interaction between the bully and a child, and the player giving the child advice.

The educational interactive narrative *Overcoming Refugee Integration with Empathic Novel Technology* (*ORIENT*) (Mascarenhas et al. 2010; R. Aylett et al. 2009; Kriegel et al. 2010) fosters social and emotional learning by enabling educational role playing. The virtual environment focuses on evoking inter-cultural empathy via conflict resolution. Players must interpret subtle cultural differences and establish relationships with characters from alien cultures to save the world from impending doom. *ORIENT* makes use of a combination of *FAtiMA* and *PSI* (Dörner 2003), a psychological model that comprehensively integrates the basic components of human action regulation, called *FAtiMA-PSI. ORIENT* is aimed at being an educational experience with the player having an avatar while *Prom Week* was developed to be a video game where the player has no direct avatar. *FAtiMA-PSI,* much like *PsychSim*,

interesting dramaturgical method for exploring new ways to confront problems (Boal and Picher 2007; Dennis 2009).

models characters with psychology (inside the minds of characters) where as *CiF* explicitly focuses on sociology (the factors between characters).

Work has been done to formalize rituals in order to generate culture-specific behavior for synthetic characters. The concept of ritual was defined and integrated into a synthetic character architecture as a symbolic social activity that is carried out in a predetermined fashion. Expressed as a goal with a pre-defined plan, rituals were used to show that synthetic characters could act in culturally specific ways. Mascarenhas et. al. formalize rituals in order to generate culture-specific behavior for synthetic characters (Mascarenhas et al. 2009; Mascarenhas et al. 2010).

One notable effort in creating a playable model of social interaction is a project aimed at comprehensively addressing the research problems around interactive storytelling. Chris Crawford's *Storytron* (formerly known as *Erasmatron*) is the result of his refocusing of development efforts exclusively to interactive storytelling (Crawford 2004). Though the domains of *CiF* and *Storytron* overlap, *Storytron* is designed to be reactive to all the events in a story world while *CiF* specializes in purely social interactions in authored social domains. To accomplish this, *Storytron* consists of a drama manager, *Fate*, a personality model, a actor action planner, a model of space and movement (consisting of abstract stages with simple spatial relations), and a set of 8 data types consisting of actor, verb, stage, prop, event, group, number, and flag types.

Storytron captures general interaction with character inclination formulae. Each inclination formula is evaluated on behalf of an actor and affects a specific inclination (or desire) for the actor. As *Storytron* is an interactive story authoring environment, the details of what inclinations exists and the exact inclination formulae are chosen by the author. Much like *CiF*'s influence rules (see chapter 3), the results of evaluating inclination formulae are summed per inclination so that all the inclinations of a character can be compared and used for planning and behavior selection.

CiF differs from *Storytron* in how patterns of action are determined and represented. *Storytron* takes a set of actor inclinations and uses a planner to produce a set of actions to fulfill a goal (all of which is subject to the *Fate* drama manager). In contrast, *CiF* only looks forward one step and determines a full a set of desires. Additionally, the plans generated by *Storytron* are grounded in executable actions. *CiF*'s desire formation process determines characters' volitions to perform a set of abstract social exchanges (which are then reified into performances as needed). Both systems reason over character motivation: *Storytron* with inclination equations and *CiF* with influence rules. Another major difference is that *Storytron* updates the world based solely on actions taken while *CiF* goes one step further; it processes the cascading social consequences when the storyworld changes.

The Restaurant Game is a minimal investment multiplayer online (MIMO) role-playing game that places players in a restaurant setting and allows them to interact through a set of in-game behaviors, world object interactionsk, and natural language via keyboard input in an open-ended environment (Orkin and Roy 2007). The player can take the role of either the waiter or customer. Through the traces of player interactions over 5000 gameplay sessions, a Restaurant Plan Network consisting of the common patterns of physical and linguistic interaction (encoded as STRIPS-style operators) was generated and evaluated. This Restaurant Plan Network implicitly encodes the social norms and social space transformation found in player interaction in *The Restaurant Game*. In comparison to social exchanges, the plans generated from *The Restaurant Game* data have primitives consisting of player affordances of objects and linguistic utterances where as *CiF's* social exchanges are primarily concerned with the impact that patterns of interaction have on social state.

Narrative generation systems (Lebowitz 1984; Turner 1994; Meehan 1976; Young and Riedl 2003; Figueiredo et al. 2008) model enough of a story world to create stories. In comparison, *CiF* does not attempt to model an entire story world. Instead it deeply models the myriad considerations necessary for a character to follow norms during social interactions. As such, *CiF* is meant to be the social reasoning component encompassed by a narrative generation or narrative oriented systems.

2 Social Science Foundation of *CiF*

As previously mentioned, many of the systems that enable characters to behave with some level of social competence take psychological and/or agent based approaches where the reasoning is done by a character and then acted upon. Another way to view this is that the decision making is centered inside a character's mind and primarily takes inputs sensed from the perspective of that character. These approaches typically use empirically tested models based in psychology such as the Big 5 trait model or appraisal. While this type of modeling works well in and is appropriate for the detailed modeling of individual agents that work towards satisfying their personality archetypes or responding to their appraised view of the world, I am interested in exploring the social "between" characters.

Currently, most systems either ignore the social "between" or relegate it to an artifact of the rational interactions among many self-serving agents. This is reinforced by many of the tools used to evaluate and gather data for such systems. Standard surveys, like those that employ the Likert scale (Likert 1932), and biometric data provide a great deal of detail about how a specific individual behaves. This line of study is mirrored by systems that model interactions between characters or virtual humans based on cognitive or psychological models that reason over competing capacities of a prescribed set of desires (Si, Marsella, and Pynadath 2009b; Marsella and Gratch 2009; R. S. Aylett et al. 2005). What they do not capture are the intricacies of complex ways people interact with respect to the different levels of cultural influences at play, common social practices, relationship expectations, and how these considerations change over time. This type of complex, holistic analysis of behavior can be seen in media like novels and movies but is not commonly experienced in games as evidenced in the previous discussion of playable social models.

CiF is an implementation of an alternate, norms-based vision of modeling what characters should be doing. This approach gives characters the affordances to reason over what desires are appropriate for the situation and then to negotiate between those relevant desires (Evans 2009). Through modeling normal patterns of social behavior with a context of general social norms, the amount of story space covered by each authoring effort is increased over that of authoring for a single social state.

CiF is a system that was designed to represent and reason about this complex, interconnected space in the form of a playable model. This system represents a different take on computational social behavior by focusing on deeper models of interpersonal behavior. This stands in contrast to the more common agent-based approaches that rely on psychology, rational decision making, or artificial life (Langton 1995). Instead of starting with a foundation similar to agent-based approaches, *CiF* is based on dramaturgical analysis

(Goffman 1959) and models character behavior primarily through socially normative pressures found in everyday life.

Goffman's concept of dramaturgical analysis views social interactions through the metaphor of a drama; actors, roles, props, setting, audience, and stage are all identified. *CiF* uses this aspect of dramaturgical analysis as a basis for encoding social interactions. Another key aspect of the dramaturgical metaphor is self-presentation, or the behavioral manipulation of how one is perceived by others. *CiF* was designed to work with other agent or psychological based systems to perform self-presentation in future projects (see the Integrate *CiF* with Other AI Systems section in chapter 8).

In a broad sense, *CiF* is an AI system that models a sphere of human behavior with the social science concept of symbolic interactionism (Blumer 1986). As a method of studying human group conduct, the common goal of both the AI system and symbolic interactionism is to use methodologically gathered data in a way that can synthesize or predict future behavior. The basic premises of symbolic interactionism as delineated by Blumer (Blumer 1986) are as follows:

1. "Human beings act toward things on the basis of the meanings they ascribe to those things."

2. "The meaning of such things is derived from, or arises out of, the social interaction that one has with others and the society."
3. "These meanings are handled in and modified through an interpretative process used by the person in dealing with the things he/she encounters."

As symbolic interactionism is an umbrella term that covers a large conceptual space, these basic premises vaguely shade the system presented in this dissertation. Since Blumer's work on symbolic interactionism, the concept has been refined to include both the external forces of social situations (what *CiF* excels at) and the internal force of psychology and human thinking (what systems based in psychology do well) (Charon 2009).

Two particular offshoots of symbolic interactionism are critical to *CiF*: ethnomethodology and dramaturgical analysis. The first, ethnomethodology, is a field of study with the goal of documenting how people understand their social world (Garfinkel 1967). In other words, it is the study of the everyday methods and practices members of a society use to make sense of their social environment. Interestingly, ethnomethodology lacks a formal theory and methodology. Instead it consists of a variety of techniques and stances toward researching how individuals act and what methods they use to establish orderly and patterned everyday social lives[8] (Garfinkel and Rawls 2002a). In contrast to studies of large aggregated data, ethnomethodology is a focused and deep look at a small group of people in a specific culture at a specific place and time. This actual grounding of circumstance allows for a more comprehensive understanding of a microcosm of a larger societal context (to which it may only be peripherally related). It is precisely this mix of flexibility and emphasis on deeply understanding a specific society that makes ethnomethodology useful to creating playable social models. Computer scientists and software engineers are proficient at making computational models of complex domains.

---

[8] "According to Garfinkel, if one assumes a world like this, composed on the one hand of individuals and their feelings and impulses, and on the other of the social norms and valued courses of action toward which they are constrained to orient, then one will agree with Parsons that social order is the result of general tendencies to comply with these norms. In order to demonstrate evidence of "underlying social structure" that produces these norms and values and constrains persons to follow them, a social scientist in Parsons' view must aggregate across large data sets in order to obtain findings of statistical significance. The result, according to Garfkinkel, is "Parsons' plenum": a world which is conceived of as essentially disorderly; a world in which order can only be discovered after, and as a result of, the application of a social scientific method.

Given this initial assumption, the sort of detailed study of particular places or social events advocated by Garfinkel makes no sense. Such studies could not yield evidence of order in the "plenum" which can only be revealed by large aggregate data sets. However, Garfinkel, for his part, makes the initial assumption that all socially recognizable actions must be produced in orderly and expected ways, and that they display their orderliness in their concrete details. Any individual attempting to pursue a goal would first have to produce recognizable social practices. Therefore, he argues, studying concrete practices in the situations in which they are produced gives the researcher immediate access to the process of constructing local orders. The certainty that order is displayed in the concrete details of enacted practices is not, or even firstly, a theoretical assumption but also something one feels when observing empirically the patterned orderliness of certain social occasions. Social occasions and their practices are generally recognizably orderly in ways that the Parson's plenum approach cannot account for. The experienced concrete orderliness of such occasions demands a theory that can account for it" (Garfinkel and Rawls 2002b, 93–94).

However, in the case of playable social models, the limiting factor is not creating a computational version of a domain but is the understanding of the domain itself. Ethnomethodology is a favored tool of sociology to determine domain information – the same information needed to create a playable social model.

A commonly used research method in ethnomethodology is participant observation (De Munck 2000). With the goal of gaining intimate knowledge and familiarity with a culture, participant observation relies on immersion through intensive involvement in a culture. By being thoroughly embedded in a cultural space, the observer transitions to a participant-observer and is able to achieve a deeper level of insight about the culture in which they participated. Participant observation has a long history in the field of anthropology going back to a study of the Zuni in the mid 1800s (Cushing 1998), and was widely used by the founders of the field's modern form including Evans-Pritchard (Herskovits 1944), Mead (Mead 1928), and Malinowski (Malinowski 2012).

One constraint in this ethnomethodology to playable social model pipeline is that each case study has its own distinct and different domain. While there may be some vague and general conclusions that hold from sample to sample, the qualities that make each social context unique and worth studying on its own are exactly the qualities that increase the difficulty in making the domain general.

The goal behind *CiF* is to represent and reason about these unique qualities within the context of the general conclusions; it is a system design to match detailed performance to both broad and specific aspects of social context. The previously mentioned constraint of ethnomethodology is what makes it a great tool to unearth the type of knowledge *CiF* needs.

Dramaturgical analysis is the particular ethnomethodological method used by *CiF* both as a basis for its system of knowledge representation and as one of the primary aspects of authoring strategy for creating content for *CiF*. To understand why dramaturgical analysis is the correct tool to use, it is useful to situate dramaturgical analysis with some of its underlying and neighboring concepts.

As human beings are individuals with their own histories, lives, and social worlds, empathy and actively thinking about the experiences of another (known as theory of mind) facilitates understanding why they behave as they do (Premack and Woodruff 1978). Perspective taking helps to frame what is seen. As we are not and cannot be completely objective observers (Bourdieu 1990, 80–97), the implicit use of perspectives during ethnographic analysis is typical. Controlling this frame of reference is a powerful tool for getting at certain types of information.

To take on a perspective of another is to attempt to understand them or, at some level, "become" them. This mental fluidity is at the core of the semiotic self which, in turn, is at the core of the dramaturgical metaphor. Put briefly, the semiotic self is a dynamic presentation of the knowledge, connections, and meanings, or signs, compiled by a person throughout their lives (Wiley 1994). In Pierce's notion of semiotics, these signs connect to one another to build recursive layers of meaning that are the foundation of the self (or even the universe itself)[9] (Corrington 1993; Bakker 2011). By consciously leveraging one's massive system of signs (aka taking on different perspective vantages: the teacher, the skeptic, the "average joe"), the large amounts of information inherent in all social interaction can be sifted; the portions salient to the taken perspective will become apparent and will be associated with one another according to the rules of the perspective.

These common semiotic patterns or perspectives frame every interaction or new bit of information we encounter[10]. They are identified as root metaphors, or overarching perspectives (Pepper 1961, 84–114,141–150). From what is seen as vital to what is ignored,

---

[9] ``It seems a strange thing, when one comes to ponder over it, that a sign should leave its interpreter to supply a part of its meaning; but the explanation of the phenomenon lies in the fact that the entire universe - not merely the universe of existents, but all that wider universe, embracing the universe of existents as a part, the universe which we are all accustomed to refer to as `the truth' - that all this universe is perfused with signs, if it is not composed exclusively of signs" (Corrington 1993, 56).

[10] These fundamental semiotic networks are continually constructed and augmented as new data is encountered. As everything encountered is placed according to the signs already in place, it essentially consists of a way a of seeing the world that is of unlimited scope, or a world hypothesis (Pepper 1961, 71–79). Interestingly "evidence and interpretation are merged" to keep the world hypothesis structurally intact and to provide consistency and coherence to the hypothesis (Pepper 1961, 79–83).

the root metaphors are distinct ways in which the world is framed. At some level, the root metaphor of an individual forms a fundamental basis of how the world is seen. However, human beings are mentally flexible and adaptable; other metaphors can be understood (to some fidelity) and willed in to play (just like you can role play or act as someone other than yourself – the recreation is never perfect but can be good for certain purposes). By taking on alternate root metaphors, a reduction of how others perceive the world can be experienced. As this perspective taking is a conscious, more objective acting effort, it leads to thinking about and analyzing what makes up the taken perspective.

A compliment to the idea of perspective taking based on the semiotic self are schemata of interpretations that, when applied to a scene[11], grant insight and meaning. Known as primary frameworks, these most primary of schemata allow "users to locate, perceive, identify, and label a seemingly infinite number of concrete occurrences defined in its terms" (Goffman 1974a, 21). According to Goffman, primary frameworks are sorted into two categories: natural (seen as "purely physical" and "unguided" by any live agency) and social (Goffman 1974b). Social frameworks give background understanding for events "guided" by a live agency for some purpose. These frames can be as simple as understanding queuing

[11] In this case, scene does not refer to a dramaturgical scene but to a more general notion of scene that can contain anything observable. Primary frameworks are a general mechanism for understanding what we encounter and are, in my estimation, intricately related (possibly even identical) to the fundamental semiotic networks we use to understand the universe.

behavior or as complicated as seeing behaviors of nations vying for natural resources through a frame of market competition.

Among the frames explored by Goffman, the dramaturgical frame is used heavily by *CiF*. Dramaturgical analysis, or the act of applying the frame of drama to a scene as a schematic of interpretation, is a useful tool to sort and connect the information of social interactions into some coherent, useful, and interconnected knowledge. It does well at representing patterns of normal social behavior and their contexts. However, the range of interaction patterns that the dramaturgical metaphor can encompass is very large: rituals, life-long performances, and simple conversations all fit into dramaturgical analysis. Both the lifelong performance of playing the unfortunate victim of fate and the short-term, routine performance of purchasing lunch from a street vendor are amenable to dramaturgical analysis. Though each example is a pattern of social behavior with social state change and contexts in which they are considered normal, the experience a player would have participating in the social behavior in each case is very different. While short-term, routine performances have been the focus of some games (particularly *The Sims 3*), focus of this work is on enabling more dramatically interesting and intense social behaviors.

Berne's theoretical framework for transactional analysis makes a clear distinction between these, and other, patterns of normal social behavior (Berne 1964). By classifying interactions into transactions, procedures, rituals, pastimes, and games, Berne categorizes patterns of social interaction by their complexity. Part of this complexity is based on three ego states or roles — parent, adult, and child — that the participants in the interaction take. If the participants both take on the expected ego states for an interaction, it is called complementary. A transaction is the basic unit of social intercourse and is based on the participants having a complementary interaction. A procedure is a set of these complementary transactions, while a ritual is like a procedure that has been stereotyped and programmed by the culture in which the participants reside. Pastimes are a series of simple, complimentary transactions that have ritualistic qualities. They are used to structure time while maximizing the social benefit to all involved. The time interval of a pastime is typically started and ended by a procedure or ritual. Examples of pastimes include: "Man Talk," "Lady Talk," "Small Talk," "Ever Been" (to a nostalgic location), and "Morning After." As the names suggest, pastimes are commonly used in social gatherings such as parties.

Social exchanges resemble Berne's notion of life games: a series of complementary ulterior transactions that are ongoing and organized into a predictable, well-defined outcome. Ulterior transactions are one of Berne's concepts that link his idea of life games with *CiF*'s dramaturgically inspired social exchanges. When the superficial, social ego states of the participants are in one interaction mode but the deeper, psychological ego states are in a different interaction mode, the transaction is considered ulterior or covert. Consider a transaction where a boss says “Could you please work Saturday?” to an employee. The common work ethic and the boss’ body language both clearly state that such a statement is an order to work over the weekend and not a request. The employee responds by saying “I had made plans for the weekend but I’ll work Saturday this one time.” Through simply hearing the words, the transaction seems to be adult-to-adult request and response. When taken in context, however, the nature of the transaction is one of parent-to-child: the demanding boss is playing a parent role and the employer is taking a capitulating, child-like role. The difference between the surface and the parallel conversations in the example makes the transaction covert or ulterior.

As this distinction allows for the decoupling of the superficial interpretation from the psychological and sociological change of interaction, ulterior transactions capture the coupling of interactions and cunning self representation that social exchanges wish to utilize. Examples of Berne’s life games are “Corner” (manipulate another so that all of their actions are wrong), “Tell Me Your Problems” (get an admission of weakness), “Sweetheart” (ridicule another in public to feel superior), and “Wooden Leg” (elicit sympathy to cover up irresponsibility or bad decisions).

Berne has assembled a catalog of games that are easily adapted to dramaturgical analysis. The cataloged games were inspired by his experience as a psychotherapist and are described in terms including a space of transactions, number of players, and psychological needs. Also described are variations in the game based on the personality and specific culture of the participants, providing us with guidance on the amount of variation to include within individual games in our social AI system.

Chapter 3

Comme il Faut

As previously stated in the related work section, the social AI system in *Comme il Faut* (*CiF*) weaves concepts from the social sciences with existing AI, computer science, and software engineering knowledge to enable the possibility of creating a richly realized interactive media experience while taming the complexities of authoring socially believable characters.

CiF (McCoy, Treanor, Samuel, Tearse, Mateas, and Wardrip-fruin 2010) uses these techniques to enable an interactive, authorable model of social interaction for autonomous agents. *Social exchanges* are the primary structure of representing social interactions in *CiF*. We define social exchanges as multi-character social interactions whose function is to modify the social state existing within and across the participants. Dramaturgical analysis (Goffman 1959) is the basis for extending the power of social exchanges from their initial form in *Façade* (Mateas and Stern 2005)—where their variation in performance was implicitly encoded in behaviors and were not reusable between characters—to their current, reusable form that supports performance variation explicitly. Through the use of social exchanges along with additional encoded social context, *CiF* is governs (or is used as inspiration for) the interactions of several virtual worlds currently in development: *Prom*

Week (McCoy, Treanor, Samuel, Tearse, Mateas, and Wardrip-Fruin 2010); *Mismanor* (Sullivan et al. 2011); the *SIREN* project in which social exchanges are used to help educators teach conflict resolution[12]; and the *Holodeck* system for enabling supporting character realism in virtual worlds. Each uses *CiF*, or a model inspired by *CiF*, to model the social aspects of their virtual worlds.

1 Design Goals

This section describes the design goals behind the creation of *CiF*. Although they are similar to the contributions in this dissertation, it is useful to state them from the perspective of *CiF* and not from the research in general. The design goal of this system is to represent and reason over compelling social situations along with the variations of the resultant behavior that arise from different personalities being placed in similar roles. However, this design goal is extremely difficult given only the tools available to computer science. Luckily, we can use knowledge from other fields to inform the design of the system. Of particular pertinence are the fields of sociology and psychology, and their subfields of micro-sociology and social psychology, along with knowledge of authoring and drama.

[12] http://sirenproject.eu/

Though the system uses concepts from sociology and psychology as the basis for models, the emphasis is not on deeply modeling or simulating social science concepts to create a fully accurate simulation of human behavior. Instead the focus is on engaging the knowledge of human behavior accrued through social science to make the characters in *CiF* more believable and compelling, particularly in their social interactions. In this way, social science is used to create a playable model of social interaction to enable a new kind of player experience.

In the same way that playable models of physics in combat games do not seek to model real physics, but rather take inspiration from physics to create a compelling experience, *CiF* does not seek to accurately model social cognition, but rather takes inspiration from a variety of social and psychological theories to create a model that underwrites compelling, playable experiences. However, I do believe that *CiF* can be a valuable tool in understanding the social science it is based on.

Authoring for interactive storytelling requires authoring a space of possible stories, while playing an interactive story is exploring one distinct story of the many possible stories that could be experienced. To make the space of stories as compelling as possible, the player should have a number of interaction options available that are consistent with the state of the story world. As each decision has impact on the story, each time the player interacts with the world the effects of past player choices need to be accounted for.

Accounting for past impacts on the story as the story unfolds leads, if handled naively, to an exponential explosion of authoring that needs to be performed. At worst every possible choice of interaction the player is presented with has to distinctly account for every possible action that could have previously taken place. If each one of these possible states of player consequence is to be explicitly detailed by an author, the burden on the authors becomes very heavy in most story worlds. The alternative, of limiting the number of choices and segregating sets of choices, makes player choice and impact feel artificially constrained. This can be clearly seen in even the best story-focused games today, such as *Mass Effect 3* (BioWare 2012), where the large number of hand-authored dialogue trees are both burdensome to produce and artificially constrained (e.g., party member relationships can only develop on the ship).

CiF represents a way to overcome these authorial challenges through building a social AI system that computationally models social space and social interaction. This makes building social play experiences more tractable by allowing the system to manage the mechanics of social interactions, which results in reducing the volume of story space to be explicitly authored and increasing the amount available for player exploration. With *CiF* and an appropriate set of content, the author of the story world now has access to what social behaviors a character should perform. In the case of *Prom Week*, the social performance information generated by *CiF* was the foundation of its gameplay.

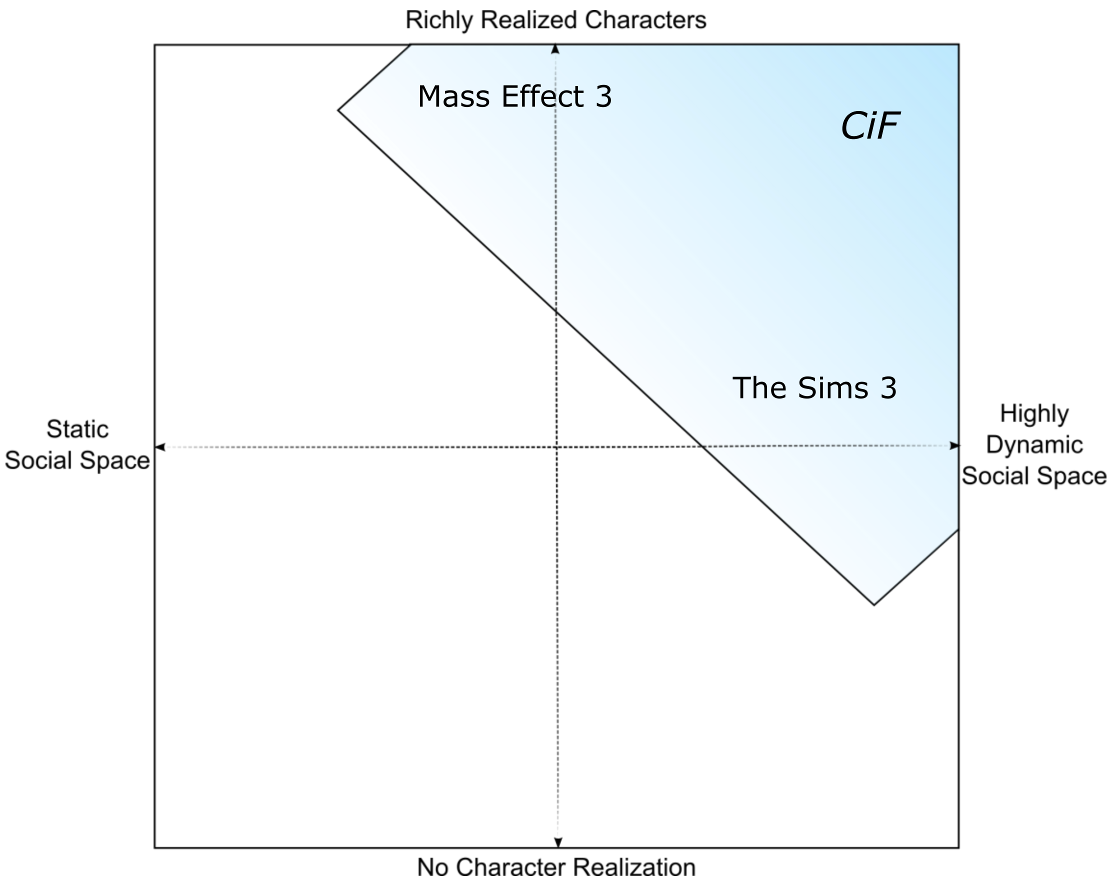


Figure 3 - This diagram shows how *Mass Effect 3, The Sims 3,* and potential *CiF*-enabled games fit in the space of dynamic social play and character realization.

Adding another layer or type of interaction in dialogue tree based games requires the previously mentioned exponential explosion in the content to be authored. Because of *CiF*'s structure and how it uses rules to encode social norms, additional social considerations can be added linearly – they are procedurally intermixed with the existing set of social considerations. For example, if there were authored rules about reciprocity and acquaintance, an author could include a set of rules about romance without modifying the rules of the two existing domains. This works nicely when the domains are highly orthogonal (if any areas of social concern can be truly orthogonal of another). However, it is up to the discretion of the author to perform a detailed mixing of existing domains, such as a romantic acquaintance. As the rules can be extremely complex, the complexity and intermixing of the social considerations made by *CiF* can fit many levels of detail.

A design goal for *CiF* was to create a playable social model that allowed for a high level of character realization in addition to supporting new levels of dynamic social play. The definition of character realization that makes sense in this context is less about polygon counts and photorealism and more about the socio-cultural binding of the characters to the storyworld. For example, CRPGs like *Mass Effect 3* have a high level of character realization as evidenced by how their behavior and dialogue match the unfolding story and social context. *The Sims 3* has a moderate amount of character realization. The characters visually match the setting (depending on the whim of the player) but their dialogue realization is loose as they speak in "simlish" and about generic topics (as specified by an icon in a dialogue bubble). With regard to social dynamics, *Mass Effect 3* gives the player choice of what NPCs to interact with and choices in the dialogue trees as affordances to social play. *The Sims 3* has much richer space of social play for the player to experience. As seen in Figure 3, *CiF* supports the best of both worlds: high degrees of character realization and a new level of dynamic social play.

Through the smart retargeting of patterns of social interaction along with additionally encoded social context, *CiF* lowers the authoring burden needed to create the social aspects of an interactive story by allowing the author to specify the rules and general patterns of how social interaction should take place. With the separation of patterns of social behavior from the norms that govern their use, authors can explicitly encode the reasoning of domains of social norms which can be reused across all social behaviors. The encoding of social norms is comprised of individual rules each of which encompasses a social consideration. Because of this rules-based encoding, additional domain knowledge can be easily added to the existing base of rules and be immediately used by *CiF*. When the rules are used in conjunction with social exchanges, the character behaviors generated by *CiF* are rich and surprising.

2 Overview of *CiF*

From a practical perspective, *CiF* primarily retargets concrete but parameterized character performances to specific social contexts (as seen in Figure 4). In order to perform this retargeting, the system needs performance patterns, a sense of what a social world is, and enough social knowledge to match the performances to the current context of the social world. Each of these broad requirements is composed of abstractions that make the reasoning over and representation of the social context and performances easier.

The *social exchange* is the most important abstraction in *CiF*; the rest of the system was formed around making use of social exchanges. Social exchanges are a method of representation meant to capture common patterns of social interaction of short duration in small group settings. Put in the dramaturgical metaphor, they can have lengths ranging from a beat to a scene. Unlike AI scripting and the static single patterns encoded by each script, social exchanges are more general and consist of a space of possible performances based on the domain of the social exchange. While the following scripts involve different

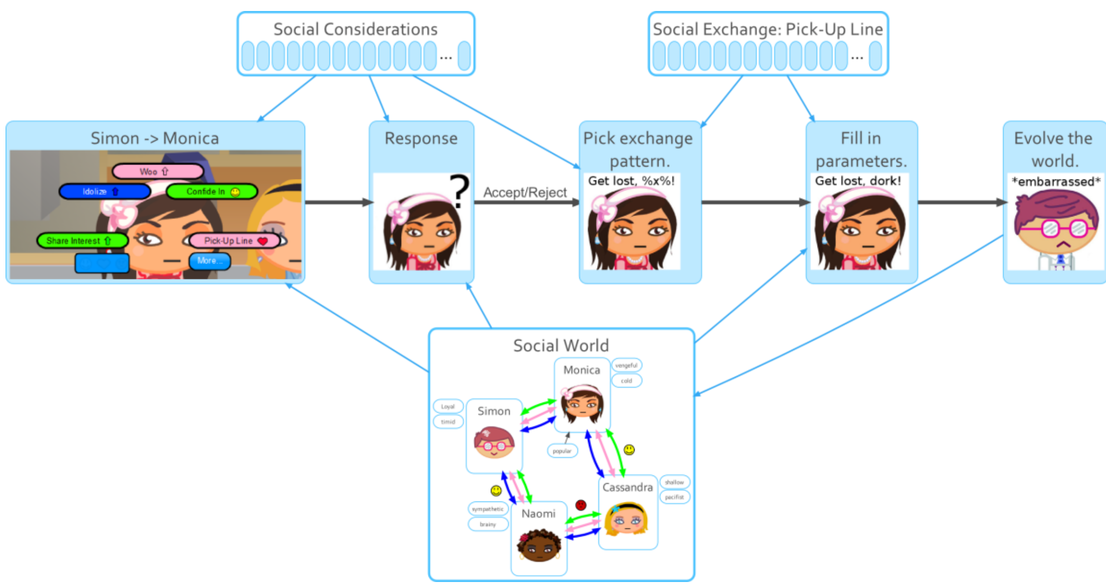


Figure 4 - This is a representation of the retargeting done by *CiF* of concrete but parameterized patterns of performance based on the social world and social consideration. This overview diagram has an embedded example of Simon interacting with Monica. The middle boxes connected by the larger dark arrows are an overview of the retargeting process. The boxes with the blue outlines at the top and bottom are the authored (in the case of social world, both procedurally updated and authored) content that the process reasons over. The authored content includes the social exchanges and their associated concrete performances.

characters and dialogue, they are included in the same social exchange, "Give Advice". The first script is between Chloe and Simon, where Simon is giving advice to Chloe about his girlfriend who is also her friend:

Simon: Hey Chloe.
Chloe: What's going on Simon?
Simon: Look, I feel obligated by social law to like you since you're my girlfriend's friend.
Chloe: Umm, okay? So don't you?
Simon: Sure! I mean... I just wanted to say I know you and Lil are good friends, so don't stop hanging out because of me. I mean, I don't have exclusive rights over her or anything. She's your friend too.
Chloe: Well, thanks for saying that. I appreciate it.
Simon: Sure.

A second script in the same social exchange still follows the domain of that exchange but the characters and performance are different:

Naomi: What's wrong Monica?
Monica: I feel like Buzz and I aren't that close.
Naomi: Don't worry about it girl. I'm sure it's just a phase.
Monica: Why would he be acting like this though?
Naomi: It's probably because I hit on him earlier, maybe.
Monica: You did what?!
Naomi: Don't get too bent out of shape about it, it was just some harmless talk.
Monica: I can't believe you. I'm out of here.

A key feature behind social exchanges is the differences and similarities between these examples. The differences represent how each script has its parameters filled in (any of the scripts could have any of the characters in the available roles – Chloe and Simon could have been replaced by Mave and Edward) and in what context a script was chosen by *CiF* as being appropriate for the social moment. For example, Chloe would be modified because she has the trait of being kind and is friends with Simon (who happens to be dating Lil). The script would be change by Monica because she has the attention hog trait and is friends with Naomi. The way a script or performance of a social exchange is chosen (or how the game plays out) is based on the characters (as was the case for Chloe and Naomi) and social state (as is the case for Monica, as that specific script calls for a pair of friends where one of which is dating a third person).

However, the examples are not totally dissimilar; the basic premise of the social exchange, giving advice, is still present. The "Giving Advice" social exchange generally represents an array of more concrete performances that adhere to the intent of a character giving advice to another character to increase their friendship. The benefit of having social exchanges encompass many different performances all with a common theme is that *CiF* can choose the most appropriate performances for the subject characters and the social world.

To facilitate the use of these patterns and to keep them applicable to any set of characters, all social exchanges in *CiF* are between an initiator and a responder. Some involve a third role called the other. Both the intent of the social exchange (the friendship improvement between the characters in "Give Advice") and the character roles are subject to social preconditions that ensure they are possible to use in the current context. Only characters that are not dating each other would be appropriate for the initiator and responder of social exchange about dating like “Pickup Line”.

CiF keeps track of the state of the social world so that it can reason over social exchanges. If the previously mentioned dating precondition of “Pickup Line” is to be evaluated, the system needs to know which characters are in dating relationships. *CiF* can model many aspects of social worlds: bidirectional networks of social attributes (like what level of romance each character has for the rest of the cast), relationships, social statuses, character traits, and social history are a sample of what the social world entails (a complete list with descriptions can be found in the Knowledge Representation section). General aspects of the social world are provided by *CiF* and are meant to be filled out by authors. This typically involves deciding what is important to the storyworld; general support for relationships exists but the author needs to decide that Lover and Boss are in the storyworld but Sibling and Best Friend relationships are not. The social world is changed through characters performing social exchanges with one another.

Social exchanges can been deemed appropriate by querying the social world. As a result, there are a set of social exchanges that are possible to perform. Just because a social exchange is possible given the social world does not mean a character would want to perform it. If Simon and Lil were dating and had a great relationship, why would either want to “Txt Msg Breakup” with the other when there are other possibilities such as “Flirt” or “Confide In”? All of these social exchanges are possible in the context. It is *CiF’s* job to determine which of these possibilities are desirable for characters to perform given the social norms of the storyworld.

The last area of authored content, social considerations, are the rules of normal social behavior. These are what *CiF* uses to determine how desirable a social exchange is to an initiating character with a target character, or responder, in mind. A shy character being less likely to “Ask Out” another character or someone is more likely to be friends with another character when a mutual enemy is involved are a couple of examples of social considerations. These social considerations are rules that capture a small corner of the social behavior in the story world. Combined, they create a space of social norms in which *CiF* can determine how desirable social exchanges are between characters.

With the three areas of content, *CiF* can go about its process of forming a ranked set of social exchanges for characters to choose from, pick a social exchange to perform, determine how the responder reacts to the social exchange, pick an appropriate pattern from within the social exchange, fill in the parameters of the chosen pattern with detail of the character and social world, and finally evolve the social world based on the social change brought about by performing the social exchange pattern. This entire process (seen in Figure 4) is repeated after every social exchange performance.

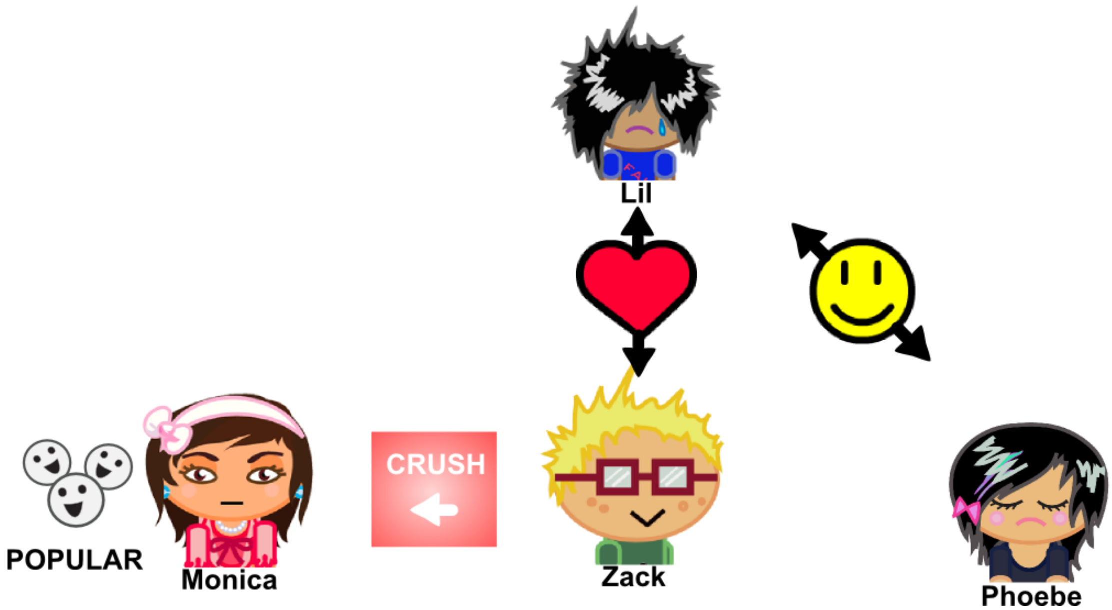


Figure 5 - This visualization of a social state shows the relationships and statuses of the characters Lil, Moncia, Zack, and Phoebe. The icons from left to right mean that Monica is popular, Zack has a crush on Monica, Lil and Zack are dating, and Lil and Phoebe are friends. This situation is dramatically loaded; if Zack acts on his crush and dates Monica, he runs the risk of having to face Lil and Phoebe's ire.

The following example from *Prom Week* serves to clarify the process used by *CiF*. Zack has been dating Lil for a few weeks and their relationship is beginning to ebb (a visualization of this social world is shown in Figure 5). Also, Zack has always had a crush on the popular girl, Monica. Lil's friend Phoebe has never had a good opinion of Zack. So, if the player decides to have Zack ask Monica out on a date, what should happen? Well, first *CiF* looks at the social world between the characters and looks at the social norms the characters abide by. Does it even make sense for Zack as a character to want to date Monica given that he already has a girlfriend? In this case, the factors of Monica being cute and popular, Zack's crush on her, and his waning relationship with Lil all contribute to his wanting to ask out Monica. Working against the potential asking out is the fact that Zack is already in a dating relationship. In this case, the authored set of social norms determines that the positives for the asking out outweigh the negatives. *CiF* has now determined that Zack asking out Monica is within the social norms of the story world.

Now, assuming the player chooses to do so, Zack asks out Monica. How will Monica respond? A process similar to seeing if it was acceptable for Zack to want to ask out Monica is performed to determine if Monica should accept or reject Zack's advances. On the side of accept is Monica's will to be liked by everyone, her being brainy (which is a trait shared by Zack), those flowers that Zack gave her in the 7th grade, and that her friends think Zack is cool. Opposing factors are Zack's awkwardness and that Zack is already dating Lil (who is so not cool). Accept wins and Monica accepts Zack's advances; they are now dating.

What should happen now? Cheating is a very dramatic occurrence in high school and should have lots of interesting consequences. Zack should be branded a cheater; Monica should be a homewrecker; Lil should be heartbroken; Monica's friends should think Zack is cooler; Lil's friend Phoebe should be very angry with Zack; and Lil and Monica should have an even more strained relationship. How many of these consequences does *CiF* recognize?

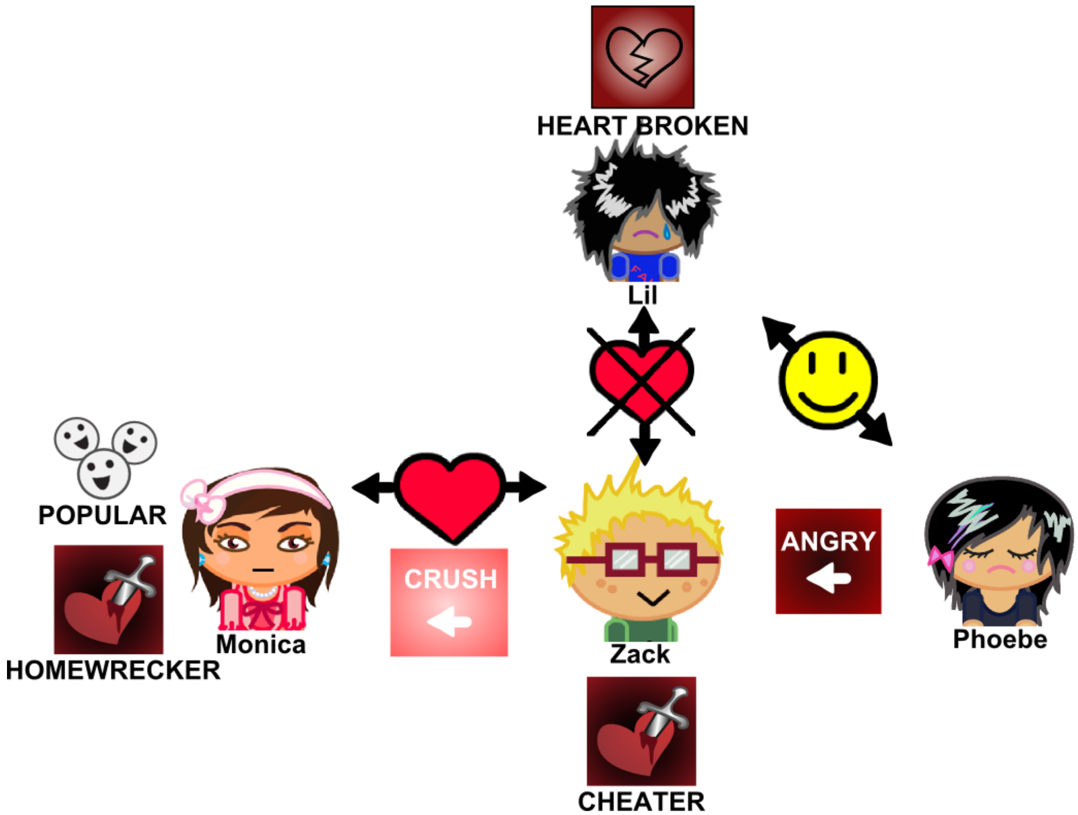


Figure 6 – This figure is a visualization of the social state showing the social consequences of Zack dating Monica. The changes in the social state (from left to right, top to bottom) are Monica is homewrecker, Zack and Monica are dating, Lil is heartbroken, Zack and Lil are no longer dating, Zack is known as a cheater, and Phoebe is angry at Zack.

All of them; *CiF* was engineered to reason over social norms, patterns of social behavior, and the fallout of social change (like cheating). Compare the initial social state visualization in Figure 5 with the state after Zack's advances on Monica in Figure 6. When used in *Prom Week*, *CiF* allows characters to form their own social goals while taking into account their current relationships, their histories with other characters, their common interests, and the normal social behavior of the story world.

To make the sophisticated social reasoning and dramatic social change real to the player, characters perform social exchanges in *Prom Week*. The following is script generated by *Prom Week* for Lil confronting Zack:

> *Lil: Zack! Is it true you're cheating on me with Monica?*
> *Zack: Oh... uh...I guess word travels fast?*
> *Lil: You freakshow! You horrible, horrible freakshow. You...*
> *Zack: No, gigababe, let me explain! Monica means nothing to me! I heart *you*.*
> *Lil: Do... do you really mean that?*
> *Zack: Of course, angel. I'll break it off with Monica, you'll see.*
> *Zack: ...eventually.*

3 Architecture

CiF consists of authored content and processes to use that content. An architectural overview of the system that shows the distinction between procedure and knowledge representation can be seen in Figure 7. As discussed in the related work and in *CiF*'s design goals, the knowledge representation used to encode the author's vision of the social world is primarily based on dramaturgical frame in Goffman's frame analysis (see the Social Science Foundation of *CiF* section in chapter 2). The other half of the playable social model is how *CiF* uses what has been authored. In the tradition of a belief, desire, and intent (BDI) model of bounded rational agents (Rao and Georgeff 1995; Bratman 1999), *CiF* uses beliefs (taking the form of authored content), calculates desires (by calculating the volition of

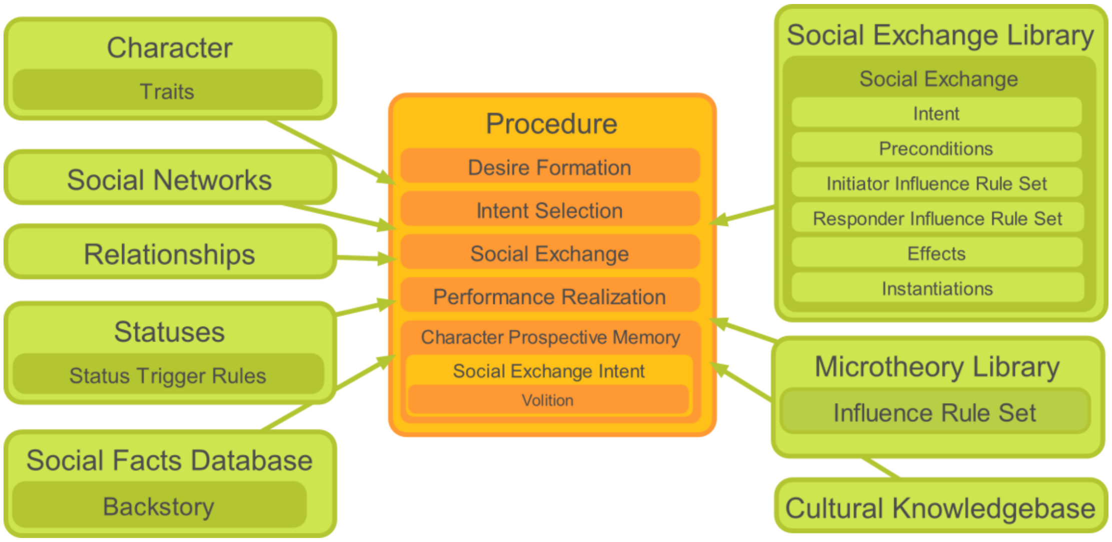


Figure 7 - The areas of knowledge representation and algorithms that comprise *CiF's* framework. The outer blocks are the areas of *CiF* that require authored knowledge and in the inner block are the algorithms that act upon the authored knowledge.

characters to play social exchanges with other characters), and forms intents (choosing a social exchange to enact).

The following Knowledge Representation and Process sections contain detailed descriptions of how *CiF* represents knowledge and processes it to create a playable social model. To help illustrate the system, examples of the authored content from *Prom Week* will be included.

4 Knowledge Representation

As discussed earlier, dramaturgical analysis uses metaphors from theater to analyze everyday life interactions: agents are a cast, actions are performances that take place in settings, objects in the world being props, and allied actors form performance teams. To this end, social exchanges represent a related set of patterns of social interaction where the exact performance and social outcome varies based on the personality-specific attributes of the characters involved; they are sensitive to the variance in performances that can be seen through the dramaturgical metaphor.

In comparison to hierarchical task networks (Erol, Hendler, and Nau 1995) and behavior trees (Isla 2005), the operators, or patterns of social behavior, in *CiF* make use of larger sets of domain knowledge to judge their appropriateness for the current context. Instead of encapsulating domain knowledge implicitly in hierarchically layered operators or behaviors using a small number of (possibly procedural) pre or post conditions, *CiF* chooses characters' behaviors based on all applicable rules in a large rulebase that encodes normal social behavior authored for a particular story world.

Representing the patterns of interaction found in these social norms has been researched at a sociological level by Goffman and as a knowledge representation problem by those in the research tradition started by Schank. Schank introduced a conceptual dependency theory as a knowledge representation tool for natural language generation (Schank 1975). Later, he introduced the constructions of scripts, plans, and themes to be used as tools to capture the domain of story-level understanding (Schank and Abelson 1977). Much like case-based reasoning (of which Schank was a major contributor), the basic unit of memory in his conceptual dependency theory was a script, or a generalized plan built from a small number of primitives. He viewed memory in the form of meaningful stories and used references to scripts to build memory; a set of references to scripts with some contextual information builds a story-like memory.

Social exchanges are similar to scripts. Both social exchanges and scripts are primitives arranged in graph-like structures to capture behavior patterns. A main difference is that primitives in scripts are meant to be general and stem from representing the underlying structures of natural language while those of social exchanges are specialized into social space transformation and are based in the dramaturgical metaphor. The reasoning over and storage of social history in *CiF* is very much like Schank's dynamic memory in that references to social exchanges along with contextual information comprise the social history of the agents. A key feature of *CiF* mechanism for storing history, the social facts database (described later in its own section), is that social exchanges given labels, such as lame or bad ass, as they are stored. This labeling becomes a primary mechanism for the recall and use of events in history.

The remainder of this section is comprised of explanations and examples of the areas of knowledge representation in *CiF*. The examples come from authored content in *Prom Week* and will be used to illustrate the data structures and their use. At a high level, there are two areas of representation. The first are the representations of what can be true in the social world. This category is designed to give a sense of social state; relationships that exist, statuses characters posses, and social history are all part of the description of a social "here and now." This notion of "what is true now" requires bounds or a space of play in the form of "what can be true". As such, this first category of knowledge representation includes an authorable range such as what relationships are in the world. Relationships, character traits, statuses, social networks, cultural knowledge, and social history are all in this category.

The second category consists of ways to reason over social state and the abstract performances found in social exchanges. As previously discussed, social exchanges are abstract patterns of social interactions that characters can use to change the social state, and that contain a set of concrete performances. In order to perform the smart retargeting of performances according to the current storyworld context, social exchanges and the aforementioned "social considerations" have the capacity to reason over the current social environment. *CiF* performs this reasoning with a rules system that consists of logically combined primitive elements that can be evaluated for truth (and, in most cases, be used to make what they represent true in the social world). These rules encode small domains of social knowledge such as if two characters are dating, they should be more likely to enact performances that increase their romantic feelings for one another. As this rule could apply to any two characters in the cast, the rules can refer to specific character or leave a slot for any character to fill. During the reasoning process, characters can be bound to variables in rules to make a generic rule applicable to a specific situation.

In the following examples, x, y, and z stand for character variables that are filled either by context or by *CiF*. As most of the areas of knowledge representation can be leveraged in *CiF*'s rule system, the predicate notation (which is shorthand for the primitive components, or predicates, that comprise rule conditions) will be used to refer to specific examples in text. The predicate notation will resemble the following relationship example: relationship(dating,x,y).

4.1 Relationships

The relationships (see Table 1) are binary states that provide detailed information about the social state of agents. *CiF's* relationships are bidirectional and are the feature of the social state that has the most impact on social exchange play. For example, *Prom Week* has three different types of relationships, including relationship(friends,x,y), relationship(dating,x,y), and relationship(enemies,x,y). Relationships are stored in a network that contains all of the relationships between two characters as an edge. As *CiF's* notion of relationships is public, binary, and considered bi-directional.

Table 1- The relationship data structure.

Relationship	
Type	Holds a label that denotes the relationship type (friends, dating,

	etc).
X	The first of two characters in the relationship.
Y	The second of two characters in the relationship.

4.2 Characters

Due to the emphasis being placed on social norms and how they guide social exchanges, the representation of each character is thin. What makes a character rich and unique is their situation in the social world and their history. This is a direct artifact of the sociological base of *CiF*; the characters are modeled as semiotic selves. How a character represents herself at any given point in time is not a function of a core and indivisible identity but is rather a fluid combination of society, history, and current circumstance.

We found while authoring in *CiF* that having some character-specific attributes that are immutable was valuable. This is where Reiss' motivational (Reiss 2008) analysis is most strongly represented as character traits (*Prom Week's* were loosely adapted from the list of traits complied by Reiss). Traits allow for more fine-grained and consistent control over how individual characters perform.

In *CiF*, the character data structure is relatively simple and serves as a place to store information related to the character (see Table 2). Static information (the character's name, gender, and locutions) is stored as class elements. A small amount of character-specific state (traits and statutes) is held by the data structure corresponding to a character. The most complicated element of the data structure is the prospective memory. It is used to store the desires (or prospects for action) calculated by *CiF*'s desire formation process.

Table 2 – The character data structure.

Character	
Name	The character's name.
Gender	The gender of the character.
Traits	A vector of the character's traits.
Statuses	A vector of the statuses held by the character.
Prospective Memory	The character's desires to play social exchanges. This is a vector of volitions (explained in the Desire Formation section).
Character-specific Phrases	The character-specific natural language generation template fill-ins (greetings, pejoratives, exclamations, etc).

4.2.1 Traits

Unlike social network values, relationships, and statuses, traits are permanent properties of a character which heavily impact social exchange play (e.g. trait(brainy,x)). In reality personality traits can change over a long enough time scale. Since *CiF* is generally used in shorter duration fiction, character traits are not able to be changed.

4.3 Statuses

Statuses are temporary, optionally directional, binary social effects that result from social exchange play. Statuses capture transitory states in an agent's mood (e.g. status(cheerful,x)), sharp spikes of emotion between agents (e.g. status(hasACrushOn,x,y)) and other salient facts (e.g. status(popular,x)). They are useful in capturing transitory but potent social situation and character states; being angry, embarrassed, or cheerful can all have major effects on a character's performance while not being permanent.

As statuses are not necessarily meant to be permanent, the status data structure includes a duration element. Unlike relationships (which are shared equally between two characters), statuses are strongly associated with a character. This is why the predicate notation of statuses always includes at least one character variable, like x. A character's status can be associated with or directed toward another character. A character, x, can pity or be angry at a second character, y: status(pity,x,y) or status(angryAt,x,y).

Table 3 – The status data structure.

Status	
Type	Holds a label that denotes the status type (embarrassed, popular, etc).
Directed Toward	The name of the character the status is directed toward.
Duration	The how long the status will be in effect after it is placed.

4.4 Social Networks

Social networks are bi-directional fully connected networks where the edge values measure the feelings between characters (see Figure 8). Examples include a romance network, which measures how interested characters are in pursuing intimate relationships with each other, and a "coolness" network which is an approximate record of how much respect characters have for one another. If x has a romance network value of 80 towards y, but *y* only has 20 towards x, the agents see their situation differently.

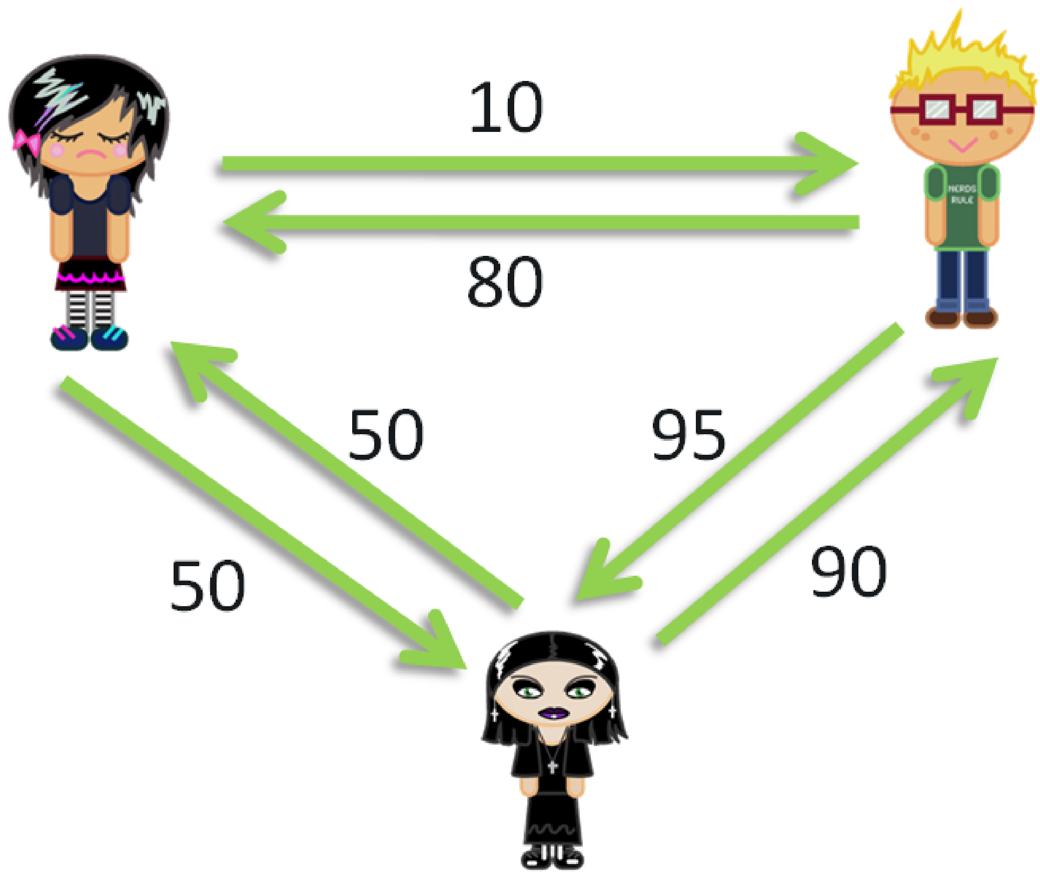


Figure 8 – This is a visualization of a social network. Each character is a node and each node is connected with bi-directional edge to all other nodes. The edges represent directional opinions over the domain of the social network. In this example, the social network domain is the characters feeling of "buddy" toward the other characters. Phoebe (at the top left) has a magnitude of 10 buddy feelings to Zack (at the top right) while he has 80 toward her.

Networks being bi-directional and distinct from relationships permits interesting (and lamentably true to life) states like relationship(dating, x, y), network(romance,x,y) is 20, network(romance,y,x) is 95, and network(romance,x,z) is 80, which translate to x and y are dating, y is head over heels in love with x, while x has fallen out of love with y but has eyes for a third character, z. These directional differences in social networks represent the internal feelings of characters toward each other (as opposed to the public nature of relationship), allow for a level of encoding of dramatic tension, and provide good hooks for rules. In the example above, even through x is dating y, x's low feelings of romance toward y would make ending that relationship more likely.

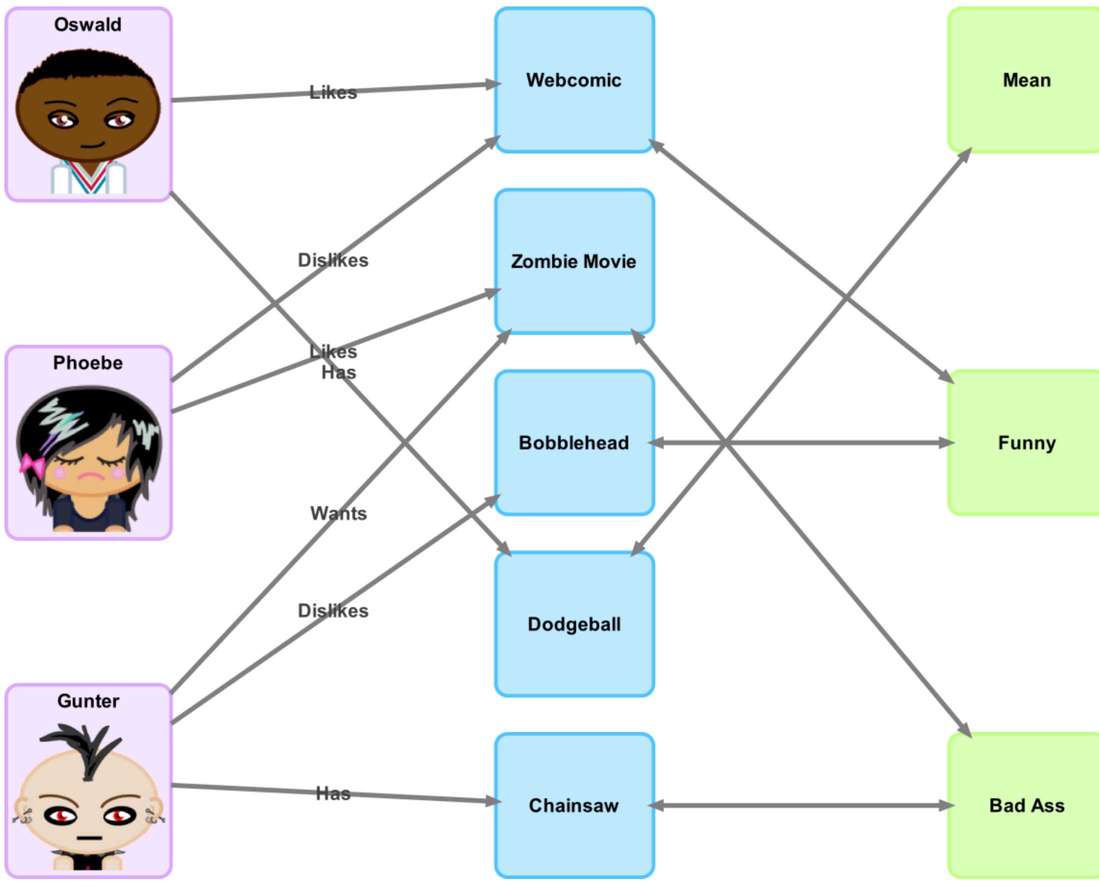


Figure 9 – An example of a cultural knowledgebase (CKB) from *Prom Week*. The left column consists of characters. The second column contains cultural items. The last column consists of cultural labels. Characters are relationally linked to cultural items (characters like, dislike, wants, has, etc. the item) and cultural items are linked to the storyworld's zeitgeist opinion of those items. In this example, an authored social exchange could leverage the fact that Oswald likes webcomics, Phoebe dislikes webcomics, and webcomics are funny in a performance in which Oswald makes fun of Phoebe for disliking something that is obviously funny.

4.5 Cultural Knowledge Base

The cultural knowledge base (CKB) is a way to further define the world that *CiF*-driven agents inhabit, providing them with a variety of topics to bond over and squabble about (see Figure 9). The design intent for creating the CKB is to have a representation of props that was sociologically rich. As props are much more than simple physical objects in dramaturgical analysis, *CiF* needs a way to understand the cultural importance of items in relationship to the storyworld. Also, the relationship to characters and the cultural items is very important. The interplay between what a prop represents to a social group and how an individual relates to that prop can widely deviate. This deviation is a great source for interactions based on social norms associated with a prop and is an interesting tool for seeding a storyworld with drama.

The CKB used for *Prom Week* has many items, including zombie movies, chainsaws, and webcomics. Every agent has one or more connections to these items, linked through the uni-directional phrases likes, dislikes, wants, and has. Gunter dislikes bobbleheads. Oswald likes web comics. Phoebie likes zombie movies. Additionally, every object in the CKB can be associated with universally agreed-upon properties in the social world (e.g. chainsaws are bad ass, dodgeball is mean). This allows for agents to interact with each other based on their individual opinions of objects in the world. The CKB can be queried to search for patterns of attitudes characters hold for objects. The follow example is of such a query:

CKB(item,(x, likes), (y, dislikes), lame)

The example illustrates how a CKB is queried to check for specific patterns in cultural knowledge. There are four parts (only one has to be specified) to a CKB query: 1) the item to look for, item 2) the first subjective label, (x, likes) 3) the second subjective label, (y, dislikes) 4) the truth label, lame. This refers to an item that x likes, y dislikes, and is universally regarded as lame, which could perhaps contribute to y's volition to poke fun at x. Conversely:

CKB(item,(x, likes), (y, likes), funny)

This statement might be used in the left-hand-side of an influence rule for a bonding social exchange, as the agents bond over their shared tastes.

4.6 Social Exchanges

As per their definition, social exchanges encompass performances salient to particular social contexts. The purpose of a social exchange can range from wanting to display coolness and sophistication to the world generally to accomplish a social state change with a particular character, like making peace with an enemy. As with all social performances, there are contexts where the exchange is appropriate and others where a break of social expectations would result. To ensure contextually appropriate performances, social exchanges contain conditions that when evaluated can determine when the social exchange is appropriate in general. One of these conditions, called preconditions, determine if the social exchange is generally applicable (characters cannot break up if they are not dating). To determine the validity of specific performances (a specific performance is known as an instantiation) in the social exchange, there are similar conditions for each instantiation. These checks are binary and any precondition or instantiation condition that fails their evaluation cannot be performed in the current social state.

If the general domain precondition check deems the social exchange possible, *CiF* has to determine the character's desire to start the exchange toward another character. Social exchanges include role specific rules that help determine desirability. These are the initiator and responder influence rule sets. These influence rule sets are used in conjunction with the authored rules for general social normalcy to determine the overall desirability of a social exchange.

Social exchanges make use of the abstraction of influence rules (described in their own section below) and an array of parameterized performances of the social exchange. Every social exchange has an initiator `i` a responder `r`, and an optional third agent referred to as the other, `o`. These roles are designed to be extremely general so they can capture many performances across a wide range of social exchanges while being specific enough to make sense when encoding performances. The initiator influence rule set and the responder influence rule set serve distinct functions when processing social exchanges. The initiator influence rule set is used in the desire formation process to determine a character's volitions to play a social exchange with other characters. The responder influence rule set factors in how the responder feels about the exchange that she was included in. In a process very similar to desire formation, the responder gets to determine how they feel about the exchange – social exchanges were designed to keep the agency of the responder intact. Each instantiation can be either a reject or accept type.

The rules, social change, and performances, when considered in tandem, provide the real encoding of the authorial intent of the social exchange – the name is simply a label that should be succinct and readily evoke the domain of the exchange. An authoring advantage of the social exchange abstraction is that additional detail can be added to the social exchange by simply adding more effect and instantiation pairs.

At the data structure level, social exchanges are comprised of an intent, a set of preconditions, influence rule sets for i and r, a set of effects, and a set of instantiations. As mentioned above, the intent encodes the change i wants to make on the social state. For example, the intent of the social exchange "Ask Out" is relationship(dating,i,r). As previousl stated, preconditions are conditions which must hold true for the social exchange to be playable. The social exchange "Breakup" has a precondition of relationship(dating,i,r); before i can breakup with r, i and r must be dating.

Table 4 – The social exchange data structure.

Social Exchange	
Name	The name of the social exchange ("Ask Out", "txt msg Breakup", etc).
Intent	The intended social change associated with the social exchange.
Precondition	A condition that must be true for the social exchange is applicable in the current social environment.
Initiator Influence Rules	The social considerations applicable to the initiator of this social exchange.
Responder Influence Rules	The social considerations applicable to the responder of this social exchange.
Instantiations	The details of a specific performance including the NLG dialogue templates and character animations. Instantiations have a condition that must be true for to be performed and how that performance changes the social world.

Next, a social exchange has a set of effects, where an effect is made up of a pair of rules called the effect conditions and the social changes, and a label marking the effect as either 'accepted' or 'rejected.' The effect conditions dictate what must be true for this effect to take place, and the effect changes outline how the social state of the world is affected based on this particular effect playing out. At a high level, an effect represents one possible trace through a social exchange. At minimum, a social exchange should have two effects—a generic effect for the case in which the game is accepted (the sum of all of the rules factoring into r's considerations was positive) and another for rejection (the sum was negative). However, through the use of effect conditions, additional considerations can be taken into account which may impact the social space in additional ways. For example, given trait(cold,i), “Break Up” may not only lead to relationship(~dating(i,r)), but could have more serious repercussions as well, such as relationship(~friends(i,r)). If multiple effects have conditions which evaluate as true, the most salient effect is chosen, with saliency being a function of which true condition rule has the most predicates.

The final component to a social exchange is a set of instantiations. An instantiation has a performance consisting of lines of dialogue, each tagged with animations that communicate state change and the justifications for the state change using hand-authored natural language. As mentioned earlier, each instantiation is associated with some social changes that reflect the performance and a condition that must be true for the instantiation to be performed.

4.7 Social History

In role performances in dramaturgical analysis as well as in semiotic view of self, an actor’s history and experiences are a major factor in all aspects of performance. *CiF’s* social facts database (SFDB) is a data structure that keeps track of the social history of the storyworld so that it can be queried for socially relevant information by *CiF*’s processes.

Figure 10 – This image is a visualization of a social facts database with entries created by a player playing *Prom Week*. The yellow text at the top of the 3 boxes is a short summary of what the social exchange was about (which can be mentioned in forthcoming social exchange performances). The initiator is denoted by the left portrait and the responder by the right. The 2 lighter blue boxes in the middle show the social change that happened as a direct result of the performance (the top box) and the social consequences (the bottom box).

The SFDB stores a context entry for social exchanges played and every trigger rule that affects social state change (see the visualization in Figure 10). Additionally, any specifically mentioned cultural item, character reference, or social exchange label (such as mean, funny, and nice to) are stored in a social exchange context entry for use in future social exchanges. Through this, agents may reason over socially complex thoughts that take into account not only their current state, but the social history that led to them possessing this state. For example, Naomi will be more interested in dating Simon if Simon has done several nice things to her recently. Relevant historical facts in the SFDB can also be explicitly referenced during performance realization, in which state change is communicated to the user through hand-written instantiations.

4.8 Rule System

CiF's rule system is the mechanism by which social reasoning is encoded. A rule detects a specific condition in the social space. When evaluated, the left-hand-side (condition) of following rule can detect when two characters have a strongly romantic dating relationship:

relationship(dating,x,y) and Network(romance,x,y) > 66

This condition is made of up distinct parts that are consistent with the previously detailed ways of representing the social state. The first is, relationship(Dating,x,y), is a simple check to determine if x and y are dating. The second is a look at x's feeling of romance to y in a social network, or Network(romance,x,y) > 66. These parts are primitives in *CiF* and can be quickly and easily checked for truth. Rule primitives in *CiF* are known as predicates.

Rules in *CiF* are Horn clauses meaning that their predicates are conjunctive (or connected by a logical "and") and each has an implication[13]. The benefits of conjunctive rules are they are easier to author (this removes the dependency of authors knowing logical operations by creating an "everything must be true" rule authoring environment) and are more efficient to evaluate. This evaluation efficiency trick is used by the Prolog logical inference programming language (Colmerauer and Roussel 1996) and is achieved by making rule evaluation deterministic. If other logical operators were used, such as "or" or disjunctions, rule evaluation would become non-deterministic and would results in a large and more unpredictable search space. The drawbacks of being limited to conjunctive rules is that a rule to capture if two characters are dating or they are friends and the first has a high level of romance toward the second could not be written as the following condition:

(relationship(Dating,x,y) and relationship(Friends,x,y)) or network(Romance,x,y) > 66

However, the space denoted by the disjunctive rule can still be represented in *CiF* in the form of two separate left-hand-sides, or conditions:

relationship(Dating,x,y) and network(Romance,x,y) > 66

relationship(Friends,x,y) and network(Romance,x,y) > 66

---

[13] Horn clauses in their definite form are disjunctions of literals with at most one positive literal (i.e. $\neg p \vee \neg q \vee r$). A logically equivalent form that is conceptually useful for influence rules in *CiF* is conjunctive and an implication ($p \wedge q \rightarrow r$). A conjunctive expression with an implication is useful in *CiF* rules as each defines a social state and, when true, implies things like a weight to a desire or that a social exchange is possible.

Rules are evaluated by *CiF* in nearly all if its processes. As it is intended that a majority of rules will contain character variables, specific characters must be bound to rules for them to be properly evaluated. *CiF*'s processes manage variable binding internally (see the Process section for full details). Every process has a different context; forming desires considers every social exchange for every combination of two or three characters while social exchange play has a determined set of characters.

To add an additional level of utility, *CiF* allows rules to be created, valuated, and evaluated external to its processes. An application that employs *CiF* can create rules that can be evaluated at any time. External rules are required to provide such a binding of characters to character variables (see the external rule code example in the Predicates section).

CiF uses rules to reason over the social world when making decisions about social exchanges. To calculate a character's will, or volition, to perform social exchanges, some rules are given a weight to aid in comparing social concerns. As seen in Figure 11, the rule data structure is used in or as a foundation of every data structure in *CiF* that needs to query the social world. The remainder of this section is a discussion of rules, the predicates that form these rules, and several ways in which rules are used.

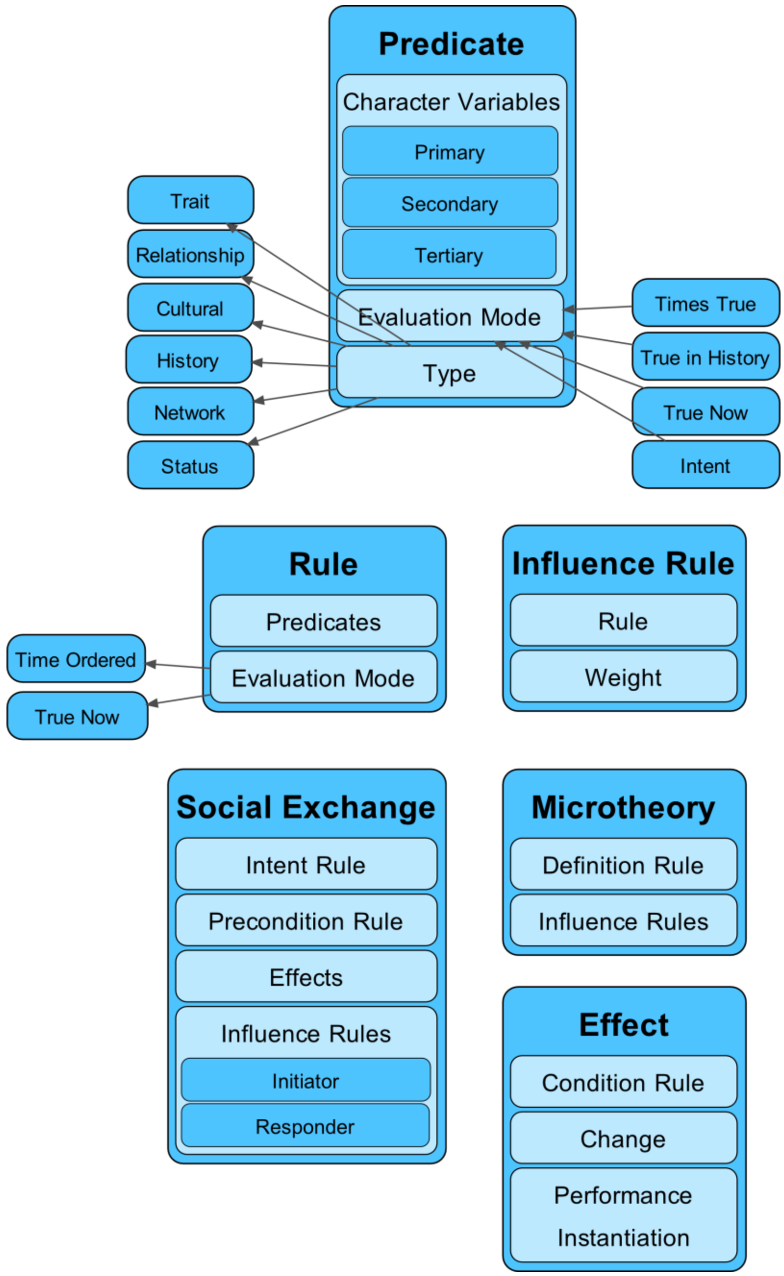


Figure 11 – The data structure schema for *CiF's* rule system.

4.8.1 Predicates

Predicates are the binding between the current social state as modeled by *CiF* and the authoring in social interaction patterns and social norms. They are representational primitives that can be evaluated for truth in a specific social state. Predicates have three areas for configuration (the areas can be seen in Figure 11). First is a set of up to three characters or character variables that will bind to characters during evaluation. Next is a predicate type corresponding to aspects of the social environment modeled by *CiF* consisting of character traits, relationships, statuses, social network values, history in the social facts database (SFDB), and cultural items in the world found in the cultural knowledgebase (CKB), which are described in detail in the Knowledge Representation section.

The final area for configuration is the details of exactly how the predicate is evaluated, or the evaluation mode. A predicate can be evaluated via a few methods. These different modes of evaluation are a key feature as they allow the predicate to capture more sophisticated concepts of social space. The following evaluation modes are supported in *CiF*:

True Now. The rule is immediately evaluated for truth. This is the default (and most common in *Prom Week*) evaluation mode for predicates. SFDB labels cannot be True Now by design.

Intent. The intent mode is used during desire formation and encoding the intended consequences of social exchanges. While their primary use is internal to *CiF*, external rules that contain Intent evaluation mode predicates can be used if a social exchange reference is passed to the rule's evaluation method.

True in History. Every predicate other than trait and CKB predicates can be evaluated in True in History mode. True in History evaluations determines if the predicate is in a change rule in any SFDB entry (primarily social exchange and trigger entries). SFDB type predicates default to True in History mode. SFDB type predicates and other types of predicates that use the True in History mode seem similar at first glance. In fact they are similar. The main difference lies in authoring use (SFDB labels are meant to capture an impression of a social exchange like "mean" or "nice"), evaluation efficiency (comparing a predicate to all predicates that have taken effect in the past is expensive), and specificity (there are circumstances where knowing if a character increased their cool network value toward another character by 33 less than four turns ago is very useful).

Times True. This mode determines how many times the predicate is true in the current social state. For example, to get the status of popular, a character needs to have three or more friends or a character could be "wicked cool" if more than four other characters have a cool network value towards that character higher than 66. Two problems are solved with this predicate: 1) *CiF* only supports 3 character variables at a time 2) this saves writing really long rule conditions. Instead of writing a long condition that looks like this:

relationship(Dating,x,y) and relationship(Dating,x,z) and relationship(Dating,x,w) and relationship(Dating,x,u)

An author could create a single predicate instead. Times True requires a couple pieces of information. The first is how many times does the social state represented by the predicate need to be true? In the above examples, this would be set to three for the Popular status and four to be Wicked Cool. Second is what character variable bindings should be held static; the accepted values are "first", "second", and "both". If the predicate in the Popular example was relationship(Friends,x,y), we could set the first character variable x to be static. *CiF* would then determine how many characters could be bound to y to make the predicate evaluate to true. The number of true bindings is then compared to the Times True number (which was three for Popular) to finalize the evaluation.

The Wicked Cool status would have "second" as the character variable binding be held as static. With a Times True predicate of network(Cool,x,y) and a Times True number of four, the first character variable x could be bound to other members of the cast to see if the number of true bindings is four or above.

Table 5 – A table of predicate types and what predicate evaluation modes they support. The last column shows each predicate's support for being valuation, or making the social state described by the predicate true in the storyworld.

Type	True Now	Intent	True in History	Times True	Valuate
Relationship	✓	✓	✓	✓	✓
Status	✓	✓	✓	✓	✓
Trait	✓	✗	✗	✗	✗
Network	✓	✓	✓	✓	✓

SFDB	✗	✗	✓	✓	✓
CKB	✓	✗	✗	✓	✗

Combinations of these evaluation types yield interesting results. For example, if an SFDB predicate is evaluated with the Times True mode, it will return how many times that particular SFDB label was encountered by the characters assigned to the predicate's roles in the past (this can be capped with a history window). Some evaluation modes can be combined. Times True and True in History can be used in the same predicate to perform detailed mining of the social history; *CiF* could find out how many times a character has been cheated on or broken up with (some storyworlds could use a "freak out" behavior if something bad happens to a character many times).

Armed with a detailed description of predicates and a general understanding of rules, rules external to *CiF* can be constructed. At a high level, the process is not complex; create a rule, fill the rule with predicates, then evaluate the rule (the right-hand-side of the rule is to be handled by the external process). The following is a code example (consisting of a snippet of Actionscript 3) of creating a rule by scratch with *CiF*'s data structures:

```
var rule:Rule = new Rule();
var pred:Predicate;
var cif:CiFSingleton = CiFSingleton.getInstance(); //A reference to CiF

//The characters for the character variable binding.
var x:Character;
var y:Character;
var z:Character;
```

```
 9:
10:    //relationship(Dating, x, y)
11:    pred = new Predicate();
12:    pred.setRelationshipPredicate("x", "y", RelationshipNetwork.DATING);
13:    rule.predicates.push(pred);
14:
15:    //[network(Romance,x,y) + 20]
16:    pred = new Predicate();
17:    pred.setNetworkPredicate("y", "z", ">" , 20, SocialNetwork.ROMANCE);
18:    pred.isSFDB = true;  //turn on True in History evaluation mode
19:    rule.predicates.push(pred);
20:
21:    x = cif.cast.getCharByName("Zack")
22:    y = cif.cast.getCharByName("Chloe")
23:    z = cif.cast.getCharByName("Buzz")
24:
25:    //Evaluate the rule with the character bindings
26:    var result:Boolean = rule.evaluate(x,y,z);
```

The first line creates a new rule instantiation that will be filled in with predicates. The third line creates a local reference to *CiF*'s singleton so character references can be gathered. Lines 6-8 create unbound character references. A new predicate (relationship(Dating,x,y)) is created and pushed in to the rule's list of predicates. Similarly, in lines 16-19 a second predicate (network(Romance,y,z) > 20) that is evaluated with the True in History mode. The character variables are bound in lines 21-23 to x, y, and z being Zack, Chloe, and Buzz respectively. Finally, the rule is evaluated with the character bindings and a Boolean value is returned.

To support capabilities like character variables, predicate types, evaluation modes, and windowed SFDB history evaluation, the predicate data structure has a rich set of class elements (or member variables in Java parlance). A detailed list of the elements used by predicate is in Table 6.

Table 6 - The elements of the predicate data structure. The names of the elements are cleaned up from the standard camel case for legibility.

Predicate	
Type	The domain of the predicate (relationship, SFDB, etc).
Primary	The first character variable available to predicate. This is used to bind a character from the rule context through evaluation.
Secondary	The second character variable available to predicate.
Tertiary	The third character variable available to predicate.
Negated	True if the predicate is negated.
True in History	A flag that indicates that this predicate for SFDB lookup. This can be applied to almost every predicate type. The predicate is looked for in the change rules of any SFDB entry.
Time Order	If the predicate is in a time ordered rule, this element denotes the predicate's time priority.
Intent	Meant for use in social exchange intent and microtheories rules, a predicate with this flag is the intended consequence of a social exchange or the desire modify in influence rules.
Trait	The specific trait referenced by the predicate.
Times True	A flag for enabling the Times True evaluation mode.
Times True Count	The number of times this predicate needs to be uniquely true if the Times True evaluation mode is enabled.
Times True Character Variable	What role slots to keep static when doing a Times True mode evaluation. Acceptable values are "first", "second", and "both."
Network Value	The scalar value related to social network type predicates.
Network Comparator	The method of comparison for social network predicate evaluation.
Network Operator	How to apply the Network Value when a network predicate is valuated.
Relationship	The specific relationship referred to by a relationship type predicate.
Status	The specific status referred to by a status type predicate.
First Subjective Link	The first character to cultural item link in CKB type predicates (i.e. Edward likes Zombie Movies).
Second Subjective Link	The second character to cultural item link in CKB type predicates.
Truth Label	The "high school zeitgeist" label of a cultural item in the CKB.
Window	How far back in the past (in *CiF* turns) to look in the SFDB if the predicate is of type SFDB or the Is SFDB flag is set. The value 0 means an unlimited window.
SFDB Label	The label of an SFDB entry to match if the predicate is an SFDB type. Labels are gists or impressions of the history entry like "mean" or "romantic."

4.8.2 Influence Rules

Influence rules are *CiF* rules where the left-hand-side is a social condition and the right-hand-side consists of a weight and intent pair. *CiF*'s processes evaluate influence rules and add the weight to a character's desire toward the intent predicates when the rule's condition evaluates to true. Intents can be any predicate type that is mutable (which means the CKB and Trait predicate types are ineligible) as intents imply changing the social world in some way. Though *CiF* supports multiple intent predicates per influence rule, that capability has not been used in any significant way. Some influence rules authored for *Prom Week* are can be seen in Table 10.

Table 7 – This table contains examples of influence rules from Prom Week.

Condition (Left-Hand-Side)	Weight to an Intent (Right-Hand-Side)	Description
status(CheatingOn,r,i)	intent(relationship(Friends,i,r)) - 15	If you are being cheated on, you want to be friends with them less.
relationship(Dating,i,r) and status(AngryAt,i,r)	intent(network(Romance,i,r) +) - 1	If you are dating someone you are angry at, your desire to be romantic with them is lessened.
status(HasACrushOn,i,o) and SFDB(Romantic,r,o)	intent(~Relationship(Friends,i,r)) + 3	If your crush has done something romantic to you in the past, you are less likely to be friends with them (in favor or being "more than friends")

Most story-focused games model a character's willingness to engage in a behavior with a simple story progression point or characteristic threshold value. To enable greater dynamism, *CiF* employs influence rule sets (IRSs) — sets of rules that influence the desires of the agents to engage in social exchanges. The right-hand-side of every rule inside of an IRS is a weight that represents how important the rule is in determining intents, where an intent is the intended change in social state after performing a social exchange (e.g. have two characters start to date). Every social exchange has two influence rule sets, one for the initiator, i, of the social exchange and one for the responder, r. Though structurally equivalent, the two IRSs have contextual differences. Weights in the initiator IRS determine both which social exchanges i is interested in playing and who to perform them with. All rules, both in all initiator IRSs and in all microtheories (discussed below) are considered and their weights tallied—the social exchanges with the highest scored weights represent the social exchanges i wants to perform most. A similar scoring mechanism is used for *r*, with one small caveat; r need only decide whether to accept or reject the proposed social exchange's intent (discussed below).

Influence rules are very similar to rules (influence rules are a subclass of rules) and can be created in much the same way. The external rule code example from the previous section can be readily changed into an influence rule:

```
var irule:InfluenceRule = new InfluenceRule();
var pred:Predicate;
var cif:CiFSingleton = CiFSingleton.getInstance(); //A reference to CiF

//The characters for the character variable binding.
var x:Character;
var y:Character;
var z:Character;

//relationship(Dating, x, y)
pred = new Predicate();
pred.setRelationshipPredicate("x", "y", RelationshipNetwork.DATING);
irule.predicates.push(pred);

//[network(Romance,x,y) + 20]
pred = new Predicate();
pred.setNetworkPredicate("y", "z", ">" , 20, SocialNetwork.ROMANCE);
pred.isSFDB = true;  //turn on True in History evaluation mode
irule.predicates.push(pred);

//intent(network(Buddy,y,x) +) + 2 (x to have y feel x is a better buddy up by 2)
pred = new Predicate();
pred.setNetworkPredicate("y", "x", "+" , null, SocialNetwork.BUDDY);
pred.isIntent = true;  //turn on Intent evaluation mode
irule.predicates.push(pred);

irule.weight = 3;
```

```
29:  x = cif.cast.getCharByName("Zack")
30:  y = cif.cast.getCharByName("Chloe")
31:  z = cif.cast.getCharByName("Buzz")
32:
33:  //Evaluate the rule with the character bindings. If it's true, add the weight.
34:  var result:Boolean = irule.evaluate(x,y,z);
```

The changes in this external influence rule example from the external rule example start with changing the rule being built from Rule to InfluenceRule (the variable name is changed from rule to irule at line 1). An intent predicate intent(network(Buddy,y,x) +) is added to irule at lines 22-25. The final difference is setting the irule's weight element to 3 at line 27. Absent from this example is the actual adding of the weight to a desire.

4.8.3 Time Ordered Rules

During the development of *CiF*, we encountered authoring situations where temporal reasoning was useful. Capturing chains of social state change in history was particularly useful. When a character has a second character do something mean to them and if a third person is mean to the second, the first character should have an increased desire to be friends with the third. This "knight in shining armor" influence rule would be impossible to capture without encoding its temporal nature. Time ordered rules are an alternate evaluation mode to rules that allows for this type of temporal evaluation.

Time Ordered evaluation mode for rules follows an alternate evaluation path from the default True Now mode. Each predicate has a Time Order property that places the predicates into time groups (the default Time Order value is 0 which means current time).

The predicates are evaluated in ascending Time Order value and are evaluated in True in History mode. The following algorithm shows how Time Ordered evaluation mode operates:

```
for (order = 1; order <= maxOrderInRule; ++order) {
   for each (pred in this.predicates) {
      if (pred.timeOrder == order) {

         //the predicate is of the current order; True in History eval
         time = cif.sfdb.timeOfPredicateInHistory(pred, x, y, z);

         //was the predicate true at all in history? If not, return false.
         if (SocialFactsDB.PREDICATE_NOT_FOUND == time) return false;

         //this predicate was true only before the last order, so rule is not true.
         if (time < lastOrderTruthTime) return false;

         //update curOrderTruthTime to highest value for this order
         if (time > curOrderTruthTime) curOrderTruthTime = time;

         //if the conditions are passed, this predicate of the rule is true; continue
      }
   }
   lastOrderTruthTime = curOrderTruthTime;
}
```

All rules with a Time Order less than 1 are evaluated without temporal ordering constraints (this is not shown in code as the predicates are evaluated in the default True Now mode). This function tolerates gaps in order, meaning a rule can have predicates of orders 0, 3, 9, 100. Gaps in Time Order values are ignored. If there are multiple predicates of the same order in the rule, they must all be true after the next lowest order and before the next highest order. Any predicate of the same order is considered true as long as all other predicates of an identical are true.

4.9 Microtheories

The power of influence rule sets is great, but if each set of rules contains repetitions of influence considerations that also apply in other situations, we have found that rule sets can become unwieldy and difficult to maintain during revisions. To address this, we have introduced the concept of *microtheories* in *CiF* to capture knowledge about social dynamics that apply across multiple social exchanges. The use of microtheories is an authoring strategy that helps tame the complexity of what is essentially a big bag of rules. The microtheory library constitutes a large repository of rules, split between dozens of microtheories. A microtheory consists of a definition and a pair of influence rule sets. The definition of a microtheory is a condition, often times consisting solely of one predicate; for example, relationship(friends,x,y) is the definition of the Friends microtheory. Only microtheories whose definitions evaluate to true in the current context are considered when calculating volitions. The rule set then provides a general understanding of what it means to be friends; the first set applies to i's considerations, the second to r's. For example, friends are more likely to get along, and less likely to become enemies, than strangers.

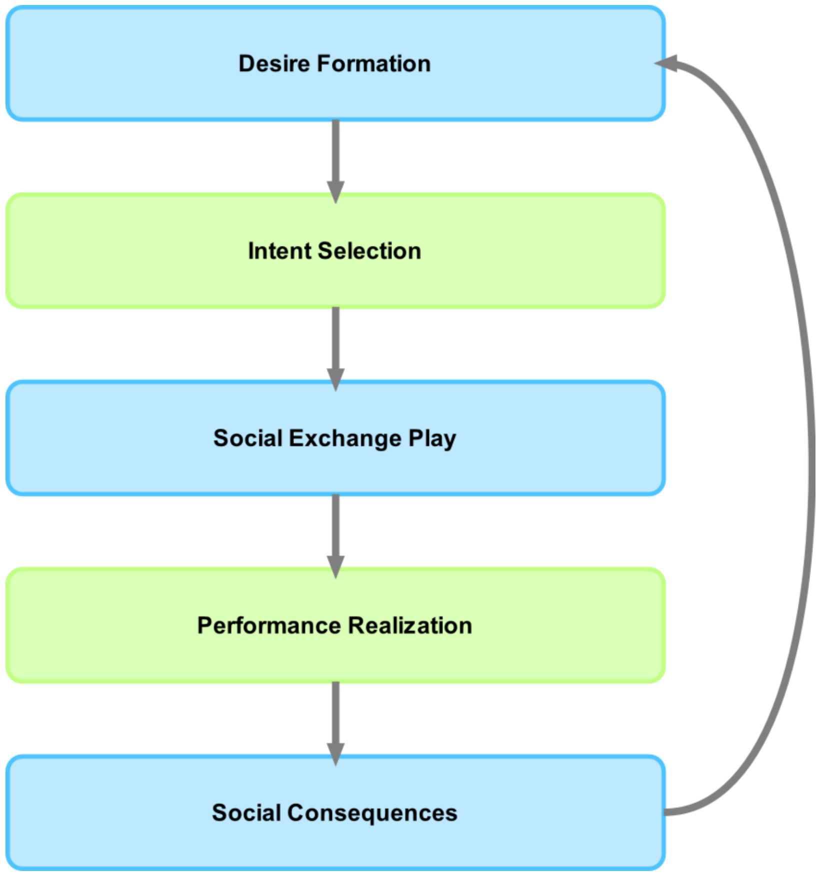


Figure 12 - A high level overview of CiF's processes.

Rules in microtheories are essentially shared by all social exchanges. This abstraction permits the initiator and responder IRSs associated with specific exchanges to focus on capturing the nuances which differentiate social exchanges from one another. For example, status(feelsSuperiorTowards,x,y) would generally negatively impact x's desire to befriend y, which is reflected its own microtheory. However, when taken in the context of the social exchange "Give Advice," it is reasonable that x would want to give advice to y, a social exchange that—given the right context—can lead two characters to friendship.

5 Process

CiF operates by looping through a set of processes (as seen in Figure 12) that leverage the authored social knowledge detailed in the previous section. To ease the deep dive into the detailed workings of *CiF*, the rest of this subsection will be a high level overview of the processes. The first process is desire formation, which determines a character's volition (or will) to perform a social exchange with roles bound to specific characters. Volition is scored by counting the weight of the true influence rules in both the microtheories with true definitions and social exchange's initiator IRS.

Next, a social exchange (with the characters bound to roles) is selected to perform. When intent is selected (discussed below), *CiF* determines how the responder chooses to either accept or reject the social exchange. This process is very similar to scoring volition in the intent formation process: the sum of true rules in both the active microtheories and the social exchange's responder influence rules set is found. If the sum is 0 or greater, the exchange is accepted. Otherwise it is rejected. Next, the process finds the most salient of the social exchanges effects to enact.

With a performance chosen and a cast known, the process of performance realization begins. Each performance has a concrete but parameterized performance script. Part of this script is a dialogue template that is filled via *CiF*'s template-based natural language generation system. When this process is complete, each character will have a fully-realized script to perform that is tailored to the cast and the current social situation.

Each social exchange effect contains a condition rule and assertions for changing the social state. The chosen effect's change is then asserted. This includes placing an entry into the SFDB to account for the performed exchange. The last step is running the trigger rules, which are rules that encode the cascading effects of social state change. Triggers are similar to effects in that they have a condition rule and assertions for social change when the condition rule is true.

After the social consequences process updates the world, *CiF* has completed one simulated turn and is ready for the next round. As this high level overview leaves out many of the details of *CiF*'s simulation cycle, the remainder of this section is dedicated to explaining each of *CiF*'s processes as seen in Figure 12.

5.1 Initialization

To facilitate easy authoring via design tools, a level of human legibility, and data portability, *CiF* reads authored content from extensible markup language (XML)[14] files. Each of the areas of knowledge representation can be written to and read from XML and *CiF* can take additional XML input at any time. Input can either be appended to what exists in *CiF* or

[14] The XML standard can be found at http://www.w3.org/XML/.

the existing content can be removed and the new XML can be read with a clean slate. *CiF* is flexible with the XML input; any major *CiF* data structure can live independently in its own XML file or broken up into several smaller files (which is what the append option is for). Input is either appended or overwritten on a per XML file basis. For example, an XML file containing a few social exchanges can be appended to the existing social exchange library followed by a complete rewriting of the microtheories. This can be accomplished with a simple function call and the use of a flag:

```
var cif:CiFSingleton = CiFSingletone.getInstance()
cif.parseCiFState(socialstateXML, false); //overwrite content
cif.parseCiFState(microtheoriesXML, true); //append content
```

Most of the data structures in *CiF* are simply read from XML values that are placed directly into data structures. The NLG dialogue content is an exception as it is stored as strings with flagged locations that are eligible to be filled in by *CiF*. After new dialogue templates are loaded, these NLG template strings need to be parsed and data structures created to hold the parsed content:

```
cif.parseCiFState(socialExchangesXML, true); //requires NLG string parsing
cif.prepareLocutions(); //parse NLG strings into data structure
```

5.2 Desire Formation

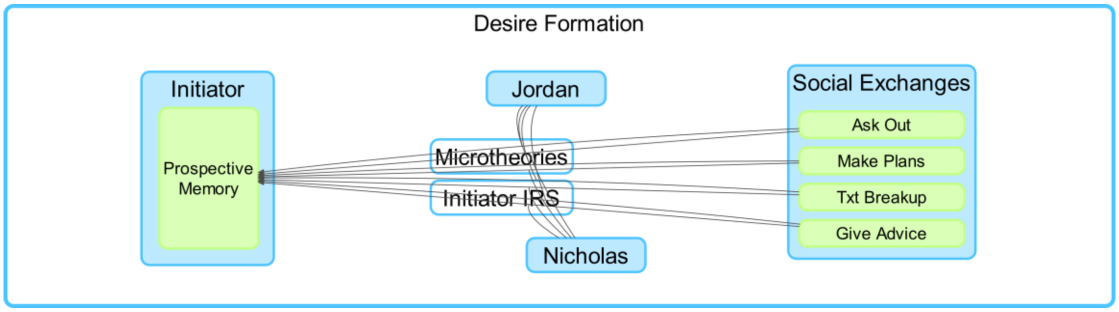


Figure 13- A view of *CiF*'s desire formation process. The character forming their desires (labeled as "initiator") considers the intent of every social exchange with respect to every other character in the cast. The influences rules in the social exchange's initiator influence rule set and the microtheories are evaluated and provide small weights to accumulate that form a desire. The result is a set of tuples consisting of a weight, a social exchange, and specific characters. These tuples comprise volition data structures and are stored in the initiator's prospective memory.

At a high level, the desire formation determines the characters' volitions to play social exchanges with each other. A set of volitions comprise a character's desire and are stored in that character's prospective memory (Ellis 1996). Each volition is a tuple of the volition score, the social exchange, and the other characters to be involved in the social exchange. In a round of full simulation, each character would form a complete set of desires. As represented in Figure 13, a complete set of desires would be to consider every social exchange and every possible set of characters that could fill the exchange's roles. In the figure, the initiator would have to consider each social exchange (assuming each social exchange has two roles slots – one for the initiator and one for a responder) with every combination of Jordan and Nicholas. The result would be eight volition data structure instantiations placed in to the initiator's prospective memory. Each of the four with social exchanges would appear twice as both Jordan and Nicholas are valid characters to fill the responder role. To complete desire formation, a complete set of desires would need to be formed with Jordan and Nicholas in the initiator role.

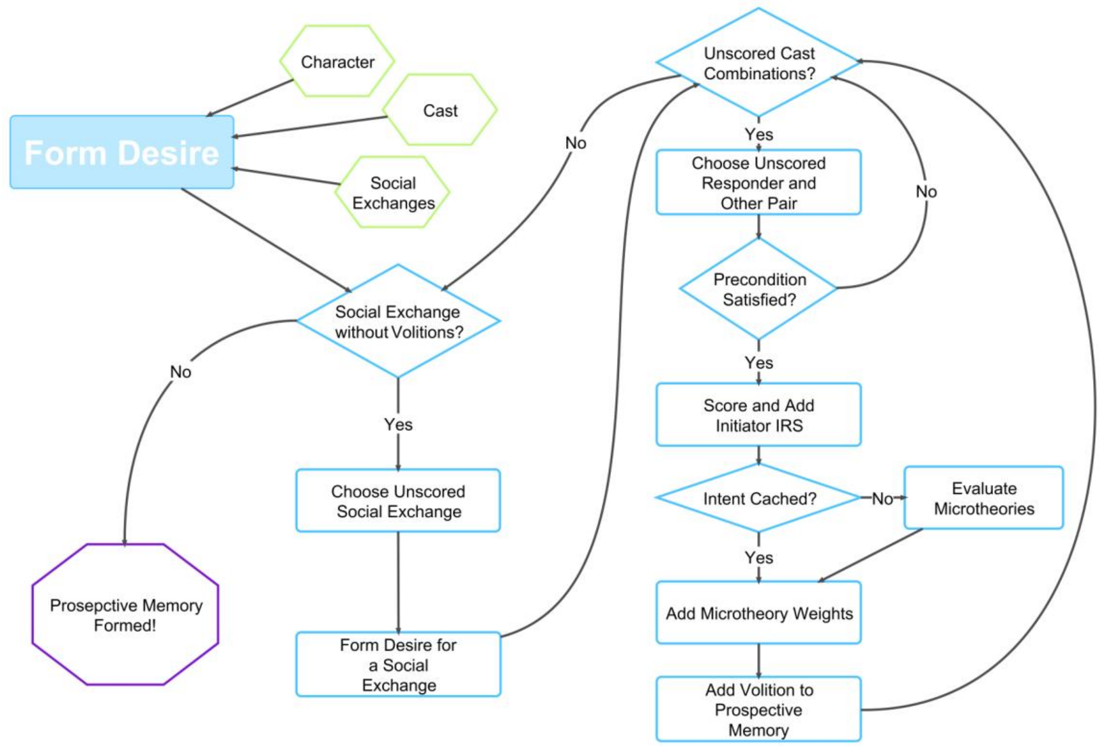


Figure 14 - A flow chart with more detail about the Desire Formation Process. The process starts at the top left Form Desire box. Hexagonal green items are input to Desire Formation. The end state is signified by the octagonal purple box labeled "Prospective Memory Formed!"

Desire formation relies heavily on influence rules in both microtheories and in the initiator IRSs in social exchanges. An accumulation of the weights of the influence rules that evaluate true is the score of a volition. In fact, the entire Desire Formation process is a way of organizing the calculation of volitions so that all necessary influence rules and character to role mappings are considered. This organization can be seen in the flow chart in Figure 14.

As evidenced by the green hexagons labeled character, cast, and social exchanges, form desire for a character is a configurable process that can take parameters. It is not always tractable or desirable for an application of *CiF* to form a complete set of desires for every character in every round of social play. *CiF* provides per character desire formation that takes a character to form desires for, a set or cast of characters to fill roles, and a set of social exchanges. By limiting what is passed to Form Desire, an application can focus only on an appropriate set of desires; a game could have only a partial cast present in a scenario or could limit the number of social exchanges available.

Because there are a potentially large number of influence rules to evaluate, *CiF* attempts to avoid re-evaluating rules. The largest collections of influence rules tend to exist in the microtheories. The first step in cutting down the number of evaluations is to avoid looking at swaths of inappropriate rules. To achieve this, the desire formation process first evaluates the definition condition of a microtheory. If the result is false, all influence rules in the microtheory can be skipped as the definition of a microtheory must be true for any of its influence rules to be valid. This mechanism is how *CiF* uses microtheories as rule libraries for each social exchange – if the definition is true, that library of rules should be included. Otherwise the rules are not included.

As microtheories encode domains of social norms, micro theories are considered in the context of each social exchange. After a microtheory is evaluated for a set of characters, the summed weight of the true influence rules are stored in the prospective memory. Because the total weight of an evaluated microtheory for a specific set of characters does not change until the social state is changed, the cached weight in the prospective memory can be used to quickly score subsequent games that need to use a previously evaluated microtheory.

As seen in Figure 14's "Score and Add Initiator IRS" block (located in the right-most column), the previously mentioned influence rules that are specific to social exchanges are used in the desire formation process. There are two sets of these influence rules: one specifically for the initiator and one for the responder. Unlike the influence rules in microtheories, the social exchange influence rules are not mirrored for use with the initiator and responder. These influence rules enable characters to have different desires for social exchanges of identical intents.

The following example will demonstrate the details of why Simon wants to perform an action that would raise Naomi's sense of romance for him (the romance network) and why Naomi rejects him so soundly. Simon and Naomi have the relationship of friends and Simon possesses the status of having a crush on Naomi. Together, these two facts activate the microtheories relationship(Friends,x,y) and status(HasACrushOn,x,y). The "friends" microtheory contains an influence rule that would detract from a character's desire to be romantic with the person he or she is friends with. However, the "has a crush on" microtheory contains an influence rule with a positive weight greater than the negative weight of the friend microtheory's rule. Furthermore, additional microtheory rules for Simon's high romance network value toward Naomi would contribute to this desire. When the weights from all true rules across all applicable microtheories are summed, the net result is that Simon wants to perform a social exchange that has the intent to raise Naomi's romantic feelings toward him (intent(network(Romance, Naomi, Simon,+)))[15].

Though there could be many social exchanges defined with this intent, the two considered here will be a physical flirt and a conversational flirt. These two social exchanges are similar by intent, but different because their preconditions, influence rule sets, and effects differ drastically. Simon's desire to perform one over the other will be determined by the two social exchanges' initiator influence rule sets in a process very similar to the microtheory's. For example, because Simon has the trait of weakling (trait(Weakling, Simon)), and there is a rule with negative weight pertaining to that trait in physical flirt, he would be less likely to want to perform that exchange. For each social

[15] Intents of social exchanges are unlike influence rules in that they have no concrete value but only indicate the direction of a social exchange. An example would be a influence rule in a microtheory would add 3 to becoming friends where as a social exchange's intent would only indicate that that they would become friends.

exchange, the net value of the weights of the true initiator influence rules add to the value of all true initiator influence rules from the microtheories to form the total desire to perform each exchange.

5.3 Intent Selection

Now that desire formation is complete and the character's prospective memories are loaded with volitions to play social exchanges, the next step is to pick a volition to act upon. Choosing a volition to enact is the goal of intent selection. In both *Prom Week* and *Mismanor*, intent selection is afforded to the player and is the core of the game play. Given the state of the storyworld and the state of the character's desires, the player chooses what happens next.

Though it is not done in the mentioned games, intent selection can readily be accomplished computationally through an intent selection policy. *CiF* was design to accommodate a simulation loop that is fully computational. *Mismanor* (Sullivan et al. 2011) built an intent selection algorithm that supported several selection policies in the form of player models to evaluate the whether and how different types of players will see different stories. To explore and test the influence rule sets, *Prom Week* had several intent selection policies constructed to provide data for machine learning algorithms.

5.4 Social Exchange Play

Social exchange play is the process that continues after intent selection is finished. It takes a selected volition and uses its contained information (character to role mappings and a referenced social exchange) to choose an NLG template and associated social change rule to pass to the next process. As seen in Figure 15, the responder has agency in social exchange play by being able to accept or reject the intent of the social exchange. This choice affects which performances contained in the social exchange are appropriate to play. From those appropriate performances, the most salient one is determined and passed along with its social change rule to the next process.

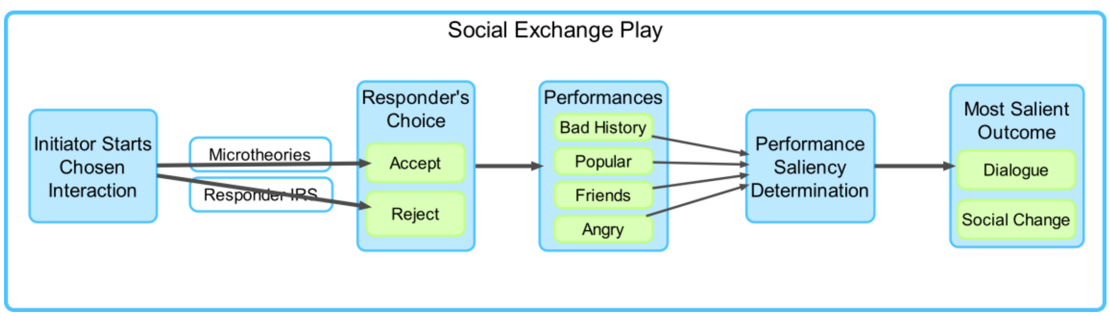


Figure 15 – This figure displays an overview of the social exchange play process. Social exchange play takes the character to role binding and social exchange chosen by the intent selection process and is responsible for selecting a natural language template and social change rule.

5.4.1 Responder's Choice

In much the same way as desire formation determines the volition for a single social exchange with a character to role mapping, the responder determines their volition to go along with the intent of the social exchange they are responding to. In this case, the microtheories and the social exchange's responder IRS are evaluated to score the responder's willingness to play along. To determine if the responder accepts or rejects the intent, the score is compared to an accept or reject baseline value. If the responder's score is greater than or equal to the baseline value, then accept is chosen. Otherwise the intent is rejected. The baseline defaults to a value of 0 but can be changed to better tune character behavior and, in the case of *Prom Week*, difficulty tuning.

The assumption this algorithm makes is that if a character as an initiator wants a specific social change (such as to become friends) to happen with another character, they will accept that same social change as a responder when the other character initiates the social exchange with that same intent. For example, consider the case where desire formation determines that Simon has a high desire to date Monica. If Monica starts a social exchange with an intent of dating and with Simon as the responder, he would accept the social exchange with the same score as his desire to date her. The only cases in *CiF* where influence rules are not reflexive are in those that are specific to a social exchange; they specifically contain sets of influence rules for both the initiator and responder are not used reflexively like the influence rules in microtheories.

Due to every performance in a social exchange being marked as either an accept or reject performance, the responder's choice eliminates all the performances that do not coincide with that choice.

5.4.2 Performance Saliency Determination

Now that either accept or reject performances are removed from the possible performances, each performance must be individually deemed appropriate for the social context. This is done by evaluating the condition rule associated with a performance (recall that Social Exchanges are coupled with a condition and a social change rule). If its condition rule evaluates as true, a performance is possible to perform. After each performance has their condition rule evaluated, the possible performances are known.

Each performance captures a specific way the social exchange can play out. As such, some are better to perform in certain social situations than others. A second comparative weighting is performed, the performance saliency determination, to rank how the possible performances fit the social context. The process examines the condition rule of each performance and scores each one. Each predicate type can contribute a different amount to the score. The default weighting scheme is that each predicate adds 1 to the score (more predicates means more restrictions on the space which, in turn, indicates more saliency). In *Prom Week,* we wanted performances feature references to player history and character relationships to be picked more often by the system. To accomplish this, *Prom Week* has a customized set of weights that counts relationship and SFDB predicate types higher and gives decaying penalties to recently enacted performances.

To illustrate social exchange play, the example started in the Desire Formation section will be continued. The next step in the saga of Naomi and Simon is to determine how Naomi will respond to his advances. The same relationship(Friends, x, y) microtheory rule that detracted from Simon's desire to perform an exchange that would increase Naomi's romance network toward Simon, also affects the responder's desire to accept the intent of the exchange. Additionally, another microtheory is brought into play by the fact that Naomi is friends with Cassie who has a crush on Simon (the microtheory with the definition of (relationship(Friends, x, z) and status(HasACrushOn, z, y)). This microtheory contains a rule that detracts from a character's desire to accept an exchange that would increase romance with the character that a friend has a crush on. This also will detract from Naomi's desire to accept the intent of the exchange.

The previous influence rules pertained to Naomi's desire to accept or reject Simon's intention to raise her feelings of romance toward him in general. The responder influence rule set and the effect conditions of the "Conversational Flirt" exchange (in conjunction with the influence rules in the microtheories) will determine exactly how she responds. For example, "Conversational Flirt" has a negatively weighted responder influence rule that examines the cultural knowledge base to see if Simon likes something that is labeled as lame that Naomi does not like.

By the microtheory and responder influence rules, it is determined that Naomi will reject Simon's advances. But the way in which she rejects him is determined by the most salient effect condition. In this case, this is an effect condition that matches the responder rule about the cultural knowledge base. The social change tied to that effect condition makes Naomi's cool network towards Simon decrease and the interaction is labeled in the social fact database as something embarrassing that Simon did to Naomi (network(Cool, Naomi, Simon,-20) and SFDB(Embarrassing, Simon, Naomi)). In *Prom Week*, effects are tied to comic book-like performance instantiations where characters engage in authored dialogue pertaining to effect conditions.

5.5 Performance Realization

Given input of a NLG dialogue template and a social change rule, the performance realization process generates a performance script that is customized to the acting characters. The social change rule is passed along to the social consequences process. *CiF* includes a template-based NLG system that fills in details as simple as names or as complex as conditional statements determined by arbitrary *CiF* rules.

Table 8 – A table that shows a comprehensive list of CiF's NLG template tags. The first column is a general description of a category of tags. The right column shows how to use the tag (in bold) and a description of what the tag generates (normal font weight).

NLG Tags	Examples and Explanations
Roles	**%i% %r% %o%** The name of the character bound to the role slot.
Role Possessive	**%ip% %rp% %op%** The corresponding character name in its possessive form.
Character Locutions	**%greeting% %shocked% %positiveAdj% %pejorative% %sweetie%** Character-specific utterances.
Pronouns	**%pron(ROLE,MALEFORM/FEMALEFORM)%**
SFDB Entry	**%SFDB_(LABEL,ROLE1,ROLE2,WINDOW)%** Inserts a SFDB reference of a previously played social exchange that matches the label, roles, and occurs in a window of time.
CKB	**%CKB_((ROLE_1,SUBJECTIVE_LABEL1),(ROLE_2,SUBJECTIVE_LABEL2),(TRUTH_LABEL))%** Inserts the name of an item that matches the specified CKB query.
Conditional Statement	**%if(ruleID,text to display)elseif(ruleID,text to display) else(text to display)%** Inserts text according to rule evaluation. There can be arbitrarily many elseif clauses.
Topics of Conversation	**%toc1% %toc2% %toc3%** Either an SFDB entry lookup or CKB item that is determined when the template is first processed and is stored to be used in the rest of the performance. A topic of conversation is metadata to the template and is specified by either a CKB or SFDB NLG tag.

The NLG templates are composed of literal and procedural pieces. The procedural pieces are denoted by pairs of % (percent signs) in a template while any text not wrapped in %s are output with no modification from the NLG system. For example, a small template, "Isn't %pron(o,he,she)% cool!?!" would be transformed into "Isn't she cool!?!" if the character bound to the other role was female. A complete list of template text options and their usage is in Table 8.

An example template that uses many of the tags can be found in *Prom Week*'s Declare War social exchange. Performance (or instantiation) number 18 includes all three roles (initiator, responder, and other) and makes use of the more straight-forward tags. The instantiation is paired with a condition rule and social change rules shown respectively below and is followed by the NLG template (the tags are bold):

Relationship(Dating,i,o) ^ relationship(Enemies,i,r) ^ relationship(Friends,r,o)
Network(buddy,r,o) + 10 ^ Network(buddy,i,o) +10 ^ SFDB(Nice,o,r) ^ SFDB(Nice,o,i)

Initiator: %greeting% %r%.

Responder: What do you want ***%i%****?*

Initiator: I'm sick of you hanging out with ***%o%****. If you keep this up I'm going to have to take drastic actions.*

Responder: What are you talking about?

Other: Hey guys, what are you up to?

Responder: Your ***%gender(i,boyfriend,girlfriend)%*** *is acting crazy. It's like* ***%pron(i,he/she)%*** *is trying to start WW3 or something.*

Initiator: I just can't handle it when you hang out with ***%pron(r,him/her)%*** *all the time.*

Other: Look, ***%r%*** *is just my friend, you are my* ***%gender(i,boyfriend,girlfriend)%****. You two are going to have to deal with that.*

Responder: Whatever happens I'm not being friends with ***%pron(i,him/her)%****.*

Initiator: Me neither.

Other: Chill the freak out. Both of you.

Initiator: Consider this war postponed... for now.

If the template was processed with Buzz as the initiator, Simon as the responder, and Naomi as the other, the template would be transformed into the following dialogue (the text generated for the tags is bold):

Buzz: ***Uhhhh Simon.***

Simon: What do you want ***Buzz****?*

Buzz: I'm sick of you hanging out with ***Naomi.*** *If you keep this up I'm going to have to take drastic actions.*

Simon: What are you talking about?

Naomi: Hey guys, what are you up to?

*Simon: Your **boyfriend** is acting crazy. It's like **he** is trying to start WW3 or something.*

*Buzz: I just can't handle it when you hang out with **her** all the time.*

*Naomi: Look, Simon is just my friend, you are my **boyfriend**. You two are going to have to deal with that.*

*Simon: Whatever happens I'm not being friends with **him**.*

Buzz: Me neither.

Naomi: Chill the freak out. Both of you.

Buzz: Consider this war postponed... for now.

The following example, Bicker instantiation 15, has a topic of conversation *%toc1%* defined[16] by the tag *%SFDB_(Romantic,r,o)%* and shows how a topic of conversation tag is used:

*Initiator: Hey **%r%**... You've really been hanging out with a lot of new faces lately...*

Responder: So?

Initiator: I dunno, it just makes me a little uncomfortable that you've been spending more time around them, than me...

Responder: Listen, you gotta get over this 'little uncomfortable' thing. When you say stuff like this, it really makes me feel suffocated.

Initiator: I just wish you'd stop giving me reasons to worry.

*Responder: What reasons to worry!? Sheesh **%sweetie%**, just quit worrying so much.*

*Responder: Although, your worries are kind of justified... Like that time when **%toc1%**...*

*Initiator: You're right **%sweetie%**. I'm sorry. I love you.*

Responder: Yeah... Totally.

[16] Topics of conversation ensure that multiple references to CKB or SFDB queries will be constant throughout a performance. If a performance has a topic of conversation, the query that will replace topic of conversation tags is ran once and used throughout the performance. This is necessary as CKB or SFDB queries randomly select an entry from all entries that satisfy the query.

With Kate as the initiator, Monica as the responder, and Nicholas as the other, *%toc1%* would be replaced with a textual reference to a past social exchange between Monica and Nicholas. In this case, that exchange would be when "%i% kissed %r% behind the bleachers after tennis practice". In these references to the past, the roles would be filled in with the context of the past exchange with respect to the characters in the current exchange (basically matching the characters of the past to the pronouns of the present). As the past social exchange had Monica as the initiator and Nicholas as the responder, *%toc1%* would be spoken by Monica as "I kissed Nicholas behind the bleachers after tennis practice". Here is the final dialogue:

*Kate: Hey **Monica**... You've really been hanging out with a lot of new faces lately...*
Monica: So?
Kate: I dunno, it just makes me a little uncomfortable that you've been spending more time around them, than me...
Monica: Listen, you gotta get over this 'little uncomfortable' thing. When you say stuff like this, it really makes me feel suffocated.
Kate: I just wish you'd stop giving me reasons to worry.
*Monica: What reasons to worry!? Sheesh **love-monkey**, just quit worrying so much.*
*Monica: Although, your worries are kind of justified... Like that time when **I kissed Nicholas behind the bleachers after tennis practice**...*
Kate: ...

5.6 Social Consequences

The final responsibility of this procedure is to record the outcome of the social exchange and true triggers in *CiF*'s social history, allowing these social consequences to be reasoned over in future social exchanges. In *CiF's* processes, only triggers and social exchanges can change this social state (applications that make use of *CiF* change the state outside of the processes). To keep an accurate history of how the social world evolves, the SFDB stores contextual entries for every social exchange played and trigger fired. An overview of processing social consequences is shown in Figure 16.

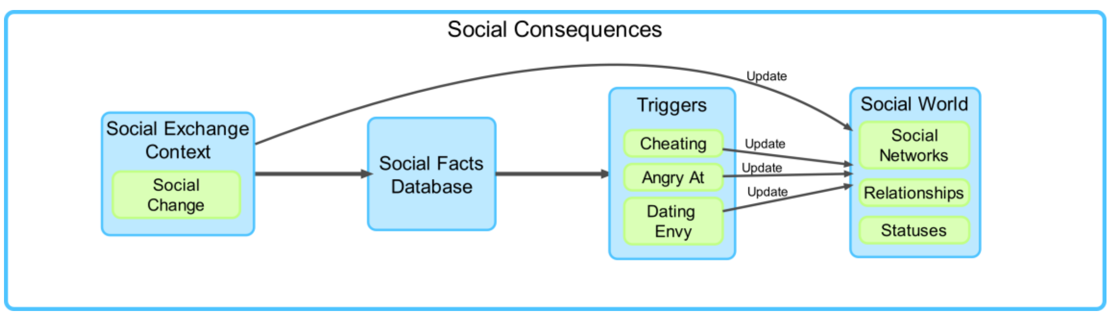


Figure 16 – This figures shows and overview of the social consequences process. Invoked after the performance realization process, it updates the social world with the social change brought about by the performance. It then processes a set of trigger rules that capture the cascading social consequences of the new social state. The social exchange and any used trigger rules are recorded in the SFDB.

The contexts stored in the SFDB contain all of the information needed to recreate or re-simulate the performances (*Prom Week* uses this capability as a save and load game mechanism). Every context stored in the SFDB has a type (like TriggerContext and SocialExchangeContext), a time stamp, and the characters involved. The social exchange contexts have the additional information about character to role bindings, the initiator's desires to play the social exchange (initiatorScore), the responder's choice (responderScore), the identifier of the performance (effectID), and the SFDB labels associated with the performance (the SFDB's labels). Examples of the XML representation of SFDB contexts are shown below:

```
<TriggerContext time="45" id="25" initiator="Oswald" responder="Buzz"/>
<SocialExchangeContext name="Embarrass Self" initiator="Phoebe"
responder="Edward" initiatorScore="5" responderScore="10" time="6"
effectID="15" other="Mave">
    <SFDBLabel type="lame" from="Phoebe" to="Edward"/>
</SocialExchangeContext>
```

5.6.1 Triggers

Triggers are rules that manage the cascading consequences caused by updating the social world with the change rule associated with the realized performance. Selected examples of triggers from *Prom Week* are in Table 9. The first two examples deal with the situation of when the person someone is dating is mean to that someone's friend. These examples are interesting because both their conditions include SFDB predicates (where the date is mean to the friend) that have a window (of 15) and are of type TimesTrue (with a value of 3). This predicate evaluates to true when the date has done something mean to the friend 3 or more times in the last 15 turns. When its condition is true, it applies the status of AngryAt to the someone directed toward the friend. The trigger was designed to chain off the effects of the first trigger; its condition also checks it see if the AnnoyedWith status is present on the someone directed toward the friend. Stacking statuses for cumulative effect was a useful design trick for authoring triggers for *Prom Week*. To ensure these two triggers would not constantly trigger one another, the first trigger's condition ensures the status applied by the next link in the chain (AngryAt) is not present.

Table 9 - Examples of triggers from Prom Week.

Condition	Change
relationship(Dating,x,z) and relationship(Friends,x,y) and ¬status(AngryAt,x,z) and SFDB(Mean,z,y,window:15,TimesTrue:3)	status(AnnoyedWith,x,z)
relationship(Dating,x,z) and relationship(Friends,x,y) and status(AnnoyedWith,x,z) and SFDB(Mean,z,y,window:15,TimesTrue:3)	status(AngryAt,x,z)
relationship(Dating,x,y) and relationship(Dating,x,z) and relationship(Dating,x,y,order:1) and relationship(Dating,x,z,order:2)	status(CheatedOnBy,y,x) and status(Cheater,x) and status(CheatingOn,x,y) and status(Homewrecked,z,y) and status(Homewrecker,z)

The third example in Table 9 is one of the most complicated trigger rules authored for *Prom Week*: the cheating trigger. It is an example of a time ordered rule. The trigger's condition is true when x is currently dating y, x is currently dating z, and x started dating y before x started dating z. The purpose of this rule being time ordered is to establish the chronology of who was dating x first. This allows us to attribute the statuses of CheatedOnBy, Cheater, CheatingOn, Homewrecked, and Homewrecker to the appropriate characters. Through assigning these statuses, the desire formation can use the statuses to capture a detailed sense of the social norms surrounding the dramatically charged social situation of cheating.

After the social exchange outcome is established and the social state is changed, the consequences of the outcome are simulated. Running the triggers over the entire cast comprise the bulk of this process by detecting change outside of the venue of the social exchange that was just enacted. Additional examples are: if Naomi just acquired her third friend then she gains the status of popular or if Simon started dating Monica while already dating Cassie, he becomes a cheater and Cassie become heartbroken.

6 Programming Example

In contrast to the complexity present in the modes of knowledge representation and processes, *CiF*'s application interface is straightforward to use. Through executing a small number of methods on *CiF*'s singleton class, an entire process loop can be simulated. The first step is to load and initialize the authored content. Next, methods to form desires, select intent, and play social exchanges (which automatically calls the social consequences process) are executed. Finally, the simulation loop can be completed by calling a method to get the realized performance script, which can be displayed according to the application. Here is a code example of the entire process:

```
//get a copy of CiF's singleton
```

```
var cif:CiFSingleton = CiFSingleton.getInstance();

//parse and initialize the authored content and gamestate
cif.parseCiFState(authoredContentAndSocialStateXML);

//desire formation over the entire cast for all social exchanges
cif.formDesires(cif.cast.characters);

//get the exchange to perform via the a weighted random selection policy
var exchange:SocialGame = cif.selectIntentWeightedRandom();

//social exchange play and social consequences (called by default).
var exchangeContext:SocialGameContext = cif.playExchange(exchange);

//performance realization process returns a processed NLG template
var script:Instantiation = cif.instantiationFromContext(exchangeContext);

//increment CiF time -- simulation loop complete!
++cif.time
```

Chapter 4

Prom Week

There were many incentives to create a game out of the affordances provided by *CiF*. Using the AI system as a tool to create an interactive experience would be a good way to drive further work on the system. Novel types of NPC interaction would be created. We would receive the potential notoriety and satisfaction that comes with creating and releasing a game. It would be a way to share the results of scientific research to the public. There would be an opportunity to validate the authorial and procedural benefits claimed by *CiF*. It could make a few people happy, sad, or otherwise emotionally impacted depending on their gameplay. While it was a risky venture (it might result in a poor game or it might be too large in scope to finish in a reasonable time), the benefits were worth the potential failure (after all, research is inherently risky). The result of this gamble was *Prom Week*, a video game that leverages *CiF* to create a new genre of game and features a new high water mark in playable social interaction.

1 *Prom Week*'s Public Presentation

The rest of this section is from *Prom Week*'s pre-release press kit[17]. It describes the public-facing overview of what the mission and features of the game are:

Prom Week is a game driven by next-generation artificial intelligence, created by students and faculty at the Expressive Intelligence Studio of UC Santa Cruz. It is a game is about social relationships — and will launch on Valentine's Day (February 14th) 2012. It will be available to play for free on Facebook and other web platforms. When the team started making *Prom Week*, the mission was to make social interactions truly playable. While games have increasingly gotten better at physical simulation, social interactions in games still tend to be scripted, with most games that provide character interaction choices using dialogue trees of some form. As a result, many games end up being about physical conflict, as the physical simulation is the only part of the system dynamic enough to enable interesting gameplay. To put it another way, just like physics simulations in puzzle games such as Angry Birds support many emergent solutions to game challenges, we want to support emergent gameplay for social interaction. That's why we refer to *Prom Week*'s underlying artificial intelligence system as a "social physics" engine. This allows us to build

[17] *Prom Week*'s pre-release press kit can be found at http://promweek.soe.ucsc.edu/PromWeekInfo.zip

on the thing that games like *The Sims* do for relatively abstract characters, but with concrete characters, speaking detailed lines of dialogue, who have particular likes, dislikes, and histories.

Prom Week supports both more casual story gameplay — in which players make choices for characters to find out what might happen next — and more strategic gameplay, with players trying to accomplish specific goals by manipulating the social environment. The moment--to--moment gameplay involves choosing what social actions characters take with one another, among those the characters most desire. What social actions characters want to take with each other and how characters responding during these social actions are determined by *Prom Week*'s social artificial intelligence system. Given a goal, such as making two characters date, and a set of characters, there are innumerable ways to accomplish this goal, all holding true to each character's personality, social context, and history. And the game remembers every action the player takes, with this history influencing character reactions and being brought up in character dialogue. This requires a very rich social simulation. The character's desires and reactions are determined by over 5,000 social considerations, rules that determine which social actions characters want to do and how they respond to social actions initiated by others. On the simple end, these considerations capture concepts such as being more likely to do something kind for someone if you're friends with them. On the complex end, the considerations handle situations like "friend jealousy" (a friend spending a lot of time with someone you're not friends with, combined with the fact that your friend hasn't spent much time with you lately, causing you to get jealous) making it more likely you'll be clingy with your friend. Additionally, the social actions play out with many dialogue and effect variations depending on the characters involved and their traits, statuses, and histories, using template-based natural language generation to create dialogue fitting the situation. And to top it all off, social actions always have lots of repercussions across multiple characters, creating a dynamic social landscape for the player to navigate.

In *Prom Week* players shape the stories of 18 high schoolers by controlling what social actions they take with one another. Each "level" of *Prom Week* is a particular character's story. Players are presented with goals that represent interesting narrative possibilities for that character in the week leading up to the high school prom. Because *Prom Week*'s social physics engine has been populated with rules that reflect the logic of high school media, making the most interesting things happen — getting characters out of their comfort zones — can require complicated manipulations of the social space. This can also be helped along by players deciding to expend an in-game resource, "story points," that can be used to predict the outcome of risky moves, or even reverse what characters would otherwise choose. This can make for more interesting stories, but comes with its own risks. Characters pushed too far, too fast may develop desires so strong that they choose to take autonomous actions that the player only has a few moves to head off. And because story points only regenerate by letting characters take the actions they desire, this resource may not be available to move things back in the player's planned trajectory, requiring some rethinking of where to take the story next. As these interactions are performed, characters refer to their pasts together. As mentioned above, some references are made to actions driven by the player earlier in the game, because what the player makes happen in this crucial week is important to the characters' lives, and *Prom Week* remembers what has taken place. Other references reveal backstory, helping the player see where characters come from and why they have certain feelings for each other. And each story culminates in the prom, with the player receiving an ending customized based on the goals she completed and the manner in which she played.

The concept of the social exchange is inspired by work from the social sciences, particularly Goffman's dramaturgical analysis and Berne's psychological games. In *Comme il Faut* a social exchange encapsulates a social action a character takes with another character (with the intention of changing their relationship) as well as how the other character responds. For example, someone might flirt with someone in order to increase romantic feelings. A character's desire to perform a social exchange is determined by social considerations, which may pull in differing directions. In addition to everything contributed at a general level by the microtheories, each exchange has its own set of rules that capture potential complexities and conflicts of feeling. At any given point, each character has a ranked list of social exchanges that he or she desires to perform (*Prom Week* has over 40). Once a player selects a social exchange for a character to perform with another, the second character, the "responder," decides whether to "accept" or "reject" the intent of the exchange (e.g., someone might reject being asked out on a date, or accept someone's compliment). Social and personal considerations are also used to determine whether he or she accepts an exchange or not. Each social exchange is associated with 20+ scenes of templated dialogue. Once the responder's general response is determined, a scene is selected and instantiated to represent how the social interaction actually takes place. The specific scene is chosen and customized based on character relationships, personalities, and histories. Each can have fallout that impacts other characters in the social network. In the end, players choose from social actions that characters desire to perform for complex reasons — and the specifics that result take similar complexities into account. The concept of the social exchange has allowed us to capture rich and dynamic patterns of social interaction without having to hand author every narrative path a player might choose.

Work on the *Comme il Faut* social physics engine has been partially supported by the National Science Foundation (award IIS--0747522 and a GRFP award) for its potential to take interactive drama and character--driven gameplay to a new level. Beyond *Prom Week*, we are working with it in a variety of under--development projects. These include the *SIREN* conflict resolution games project, for which UC Santa Cruz is collaborating with a number of European Union partners, with UCSC professor Arnav Jhala and postdoc Reid Swanson taking the lead. Another example is the *Mismanor*, in which a modified version of *Comme il Faut* is being created that enables emergent solutions to relationship--oriented quests for computer role-- playing games, an effort led by UCSC PhD students Anne Sullivan and April Grow. It is also being extended in early--stage projects involving natural language generation and cultural training. As research in this area continues to develop, we expect that social physics will help enable new genres of gameplay for entertainment, education, and a wide variety of other goals.

2 Experiment in Playable Models

The main focus of *Prom Week* was for it to serve as an evaluation via implementation of *CiF*. By going through the process of proposing an architecture, constructing playable demos, refining the AI system, and releasing a complete game, both *Prom Week* and *CiF* reached a level of power and refinement that would have been difficult or impossible to achieve if only a proposed architecture or demo was created. This complete design process is what the fields of interactive narrative and story generation are currently lacking (Horswill 2011). There exist many proposed architectures, AI systems, methods of knowledge representation, and general ideas over solving problems in the space of computational aspects of narrative and story (Tanenbaum 2011; Spierling et al. 2008; Sander 2008). There is a drop off in number of these theoretical approaches to actual systems built. Even less of these instantiated systems are validated through the creation of a demonstrative application or evaluative study. The final, and likely most severe, decline is from those systems that make it to demonstration to being used to create a fully-playable interactive narrative experience. This final step is extremely important in that it forces the underlying system to consistently produce quality results via the realization of design details.

While it certainly achieves the generic goal of being a fully-playable experience based on a research AI system, *Prom Week* is an in-depth test of *CiF*. Particularly, *CiF* as an authoring paradigm and as a computational model of social interaction are being tested through game design and game play. This test takes a few distinct forms where the first is a proof via existence or implementation – *CiF* nominally works as *Prom Week* exists as a game that definitively includes social interactions as a primary affordance. Players have experienced *Prom Week* and have been subject to a social storyworld simulated by *CiF*. This chapter's description of *Prom Week's* design reinforces this first type of evaluation. The other types are more traditional quantitative and qualitative evaluation and are discussed in chapter 5.

3 Experimental Game and Enjoyable Media Artifact

Personal and development team goals for *Prom Week* were to create an enjoyable play experience and to explore what the game could be through experimentation. While these goals are subjective and their completion is difficult to concretely ascertain, there are strong indications that *Prom Week* has achieved them. As an experimental game, *Prom Week* was a finalist for the technical excellence category in the Independent Games Festival at the Game Developer's Conference[18] and was invited by IndieCade to participate in their E3 showcase[19]. As the Independent Games Festival and IndieCade are seen as bastions of video game innovation, being selected for these events is a strong indicator of *Prom Week* being a successfully executed experimental game.

Other indications of success with respect to these goals are through the reviews and internet dialogue about *Prom Week*:

"Play this. It genuinely deserves the often-abused term 'groundbreaking.'" [20]

"...a notable advance in the state of the art of interactive narrative design."[21]

"...making social interactions into a playable experience is a revolutionary idea."[22]

[18] http://www.igf.com/02finalists.html
[19] http://www.indiecade.com/2012/indiecade_at_e3/
[20] http://www.zestforlifezine.com/reviews/prom-week-computer-game/
[21] http://playthisthing.com/prom-week

"a magnificent technical achievement... should be played by both gamers and designers with interest in increased social complexity in games."[23]

"The Next Angry Birds?"[24]

"After the grim social strategies I'd been considering, did I deserve to be Prom King? ...now I feel bad and impressed, and want to play it all over again."[25]

"...one of the more interesting social-gameplay systems out there."[26]

"While Prom Week could be played as a puzzle game, the narrative takes over almost immediately. ...The genius of Prom Week [is how] it swiftly moves beyond the mechanical, beyond the ludic, to the personal and emotional."[27]

"Prom Week's style of storytelling is more natural than The Sims. ...characters speak in text-based English."[28]

4 Basic Flow of Interaction

After being introduced to *Prom Week* via a couple of tutorial levels, players are invited to experiment with the game's affordances on their own. At a high level, the flow of

[22] http://www.kevinrothermel.com/blog/2012/2/15/prom-week-is-now-available-on-facebook.html

[23] http://www.pastemagazine.com/articles/2012/02/prom-week-review-browser-facebook.html

[24] http://news.discovery.com/human/blog-prom-week-social-game-120214.html

[25] http://www.rockpapershotgun.com/2012/02/16/impressions-prom-week/

[26] http://emshort.wordpress.com/2012/03/12/games-from-gdc-2012/

[27] http://alastairstephens.com/?p=87

[28] http://www.newscientist.com/article/mg21328555.100-be-school-prom-queen-in-social-physics-game.html

Figure 17 – A diagram showing the basic game play states in *Prom Week* and the affordances available to the player at each state. The states are shown to the left in blue and connected by arrows. To the right of each state are boxes representing what affordances are accessible to the player at that state. Green represents options, orange options cost story points, and purple are game options that do not interact with the social world of *Prom Week*. The options that are colored are available in the corresponding game state.

interaction consists of a small number of steps (an overview can be seen in Figure 17). Initially, the game is in a state where no characters are selected and only a small number of

UI elements are visible (game configuration, camera controls, and the bottom left menu). When the player selects a character, they enter the Initiator Selected state in which a social summary or the initiator's information can be accessed.

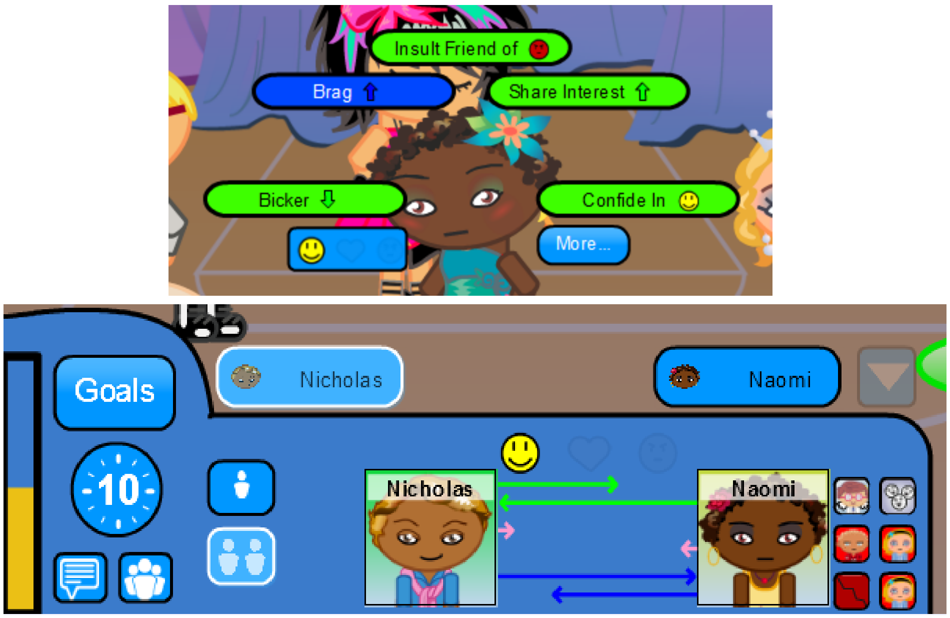


Figure 18 – This figure shows the UI elements seen in the Responder Selected state. The top image shows the ring of social exchanges (each of which has an iconic representation of the social exchange's intent). The bottom image shows many of the basic UI elements in prom week. The bar to the left is a meter that displays how many story points the player has accumulated (described below). Next to it is the story goal interface, the level timer (which can be clicked to end levels), the notification window toggle (the talk bubble icon), and the social state explorer. Near the top are icons that show the selected initiator (Nicholas) and the responder (Naomi). Below is a toggle between viewing details about the individual character (the icon with a single person) the social minimap (the icon with two people). To the bottom right is the social minimap.

Next, the Responder Selected state can be entered by selecting a second character. This gives the player access to a social "minimap" showing relationship details between the initiator and responder, the ring of social exchanges, and the option to unlock additional social exchanges (see Figure 18). The ring of social exchanges that appears over the responder's avatar represents the desires of the initiator. *Prom Week* examines the volitions stored in the initiator’s prospective memory and picks out the 5 most desirable toward the responder. If there are less than 5 aimed at the responder, the social exchange button ring shows the ones available. If there are more than 5, the surplus exchanges are placed in a dropdown selection box (appearing near the ring) in order of descending desirability. These less desirable social exchanges can be unlocked by an in-game currency called story points. The lower the volition score associated with the social exchange, the more story points it will cost to unlock.

Story points allow the player to “break” the simulation. Unlocking social exchanges that are not the most desired by the initiator is one way. The player can purchase information about what factors the responder will consider when deciding between accept and reject. Through spending a moderate amount of story points, the responder’s choice (along with how strongly that choice has been made) will be revealed before the social exchange is performed. After this information is revealed, the player can flip the responder’s decision at a large cost. Past the initial allotment, the player can gain more story points by playing social exchanges that the characters desire (those in the ring of social exchanges).

Figure 19 – A screenshot of the UI element that is made accessible after a social exchange is selected. There are four options available; two are free ("Intent" and "Motives") while the remaining two cost story points ("Factors" and "Results").

The Social Exchange Selected state is entered when the player chooses a social exchange from the ring or purchases one from the drop down menu. Two new buttons, one with the "play" icon and one with the name of the social exchange, appear between the initiator and responder buttons at the bottom (see Figure 19). The bottom button will open a panel with several new options; some are free and others cost story points. The player can freely see the initiator's intent, willingness, and motivations to play the selected social exchange. The player can spend story points to unlock the responder's factors, see what the responder will choose if the currently selected social exchange is played, or to change the responder's decision.

After the player initiates the social exchange by either clicking on the play icon or clicking on the selected social exchange a second time (which is highlighted in the ring of social exchanges), Social Exchange Started state is entered. In this state, the player is limited to configuring the game and clicking the screen to make the performance progress.

After the performance is complete, the next state is entered automatically. The next state, Results Window Displayed, shows a summary of the changes resulting from the performed social exchange. This state shows the characters involved, a sentence-length summary of the performance (which can be brought up later in gameplay), a UI element to show player detailed knowledge about the performance, icons that can post the performance to social media sites, and a "continue" button to close the UI and transition out of this state. After this state is exited, a round of gameplay is complete and the game transitions back to the No Character Selected state.

5 The Fictional World of *Prom Week*

Mirroring pre-existing media sources (like *Mean Girls*) and our own experiences with high school life and proms, *Prom Week* captures and portrays teen drama and hallway politics. In this section, the fiction (Juul 2005) of *Prom Week* is discussed to provide a foundation for more thoroughly understanding how *Prom Week* and *CiF* relate.

5.1 Setting

As the media and personal inspirations for the storyworld vary in location, *Prom Week* takes place in a generic American high school – it could be in any slightly-affluent, primarily Caucasian, medium-sized population center. As the content authored for *Prom*

Week in *CiF* is a broad view of fictional high school social norms, the behavior of the characters is of the same level of detail. Effectively, the setting is downplayed. The generic high school setting removes many of the specific details that would mire the game in the fictional world. Removing details, like the name of the gym teacher or specific problematic parents, allows the characters to emphasize the social world (and the knowledge represented by *CiF*) in their performances; the story remains concentrated on the relationships of the students and does not get lost in the setting. The concrete foundation of the setting is that *Prom Week* takes place in the school week immediately before (and including) the prom.

5.2 Cast

Consisting of 18 high school students, the cast of *Prom Week* was designed to have an interesting sample of archetypical high school students while loading the social world with potential for drama. As a holdover from the earlier design ideas of "Goths vs Emos", the cast has heavy representation of those two social groups and has since grown to encompass other stereotypical high school social groups like jocks, preps, performance arts geeks, computer nerds, punks, and skateboarders.

As they were created to produce a milieu rife with dramatic potential, the characters were authored with complex relationships with one another. In the initial social state of

Prom Week, characters begin with enemies, friends, and dates. Each relationship has varying degrees of subjective quality that correspond to their social network values toward each other. The characters are designated with statuses such as being popular or lonely. The characters each have a static set of immutable traits that serve to flavor the desires and performances of a character to make her unique. A sample of the *Prom Week* cast can be seen in Table 10.

Table 10– This table shows the descriptions of five of the 18 characters in Prom Week. Each character in Prom Week has between three and five traits, a descriptive biography, and is situated in the social world via relationships, social networks, backstory (SFDB entries from before the player is introduced into the world), and statuses.

Character	Traits	Bio
 Monica	Cold Attention Hog Sex Magnet Vengeful	Monica is a manipulative Machiavellian mastermind. Boys want her, girls want the thing she has that the boys want. She rules this school with an iron fist (fashionably manicured, of course). But smiling out from nearly every page of the yearbook, viciously stepping on the people who get in her way, in her darkest nights, Monica can't help but wonder if she's missing something from life...
 Gunter	Outgoing Confident Oblivious Ripped	Recently transferred from somewhere in Germany, Gunter is of indeterminate age and questionable intelligence. Most of his new classmates are secretly terrified of his ripped body, punk outfits, and sometimes shocking European sensibilities, although Gunter himself has not seemed to notice their reactions. Can Gunter find a place at school in time to be one of the gang by *Prom Week*?
 Zack	Arrogant Stubborn Outgoing Honest	Zack is a geek and proud of it, flaunting his l33t skillz and pop culture references at every opportunity. If he does okay in his classes it's mostly because he copies off his friend Simon: he'd much rather watch pirated anime than actually study. His single-minded stubbornness and acerbic personality have made him a lot of enemies, but if he does put his mind to something, he'll do whatever it takes to get what he wants.
 Naomi	Sympathetic Jealous Kind Brainy	Naomi, head of the prom committee and determined to make it the best prom ever, would seem to have the perfect life: dating star quarterback Buzz, top-notch SAT scores, and a kindness that's earned her friends across the social spectrum. But she can't stop worrying about the future, and is terrified that with one false move she

		might lose everything.
 Chloe	Outgoing Kind Smoother Talker Insecure	Something of a social butterfly, Chloe views herself as the great bridge, connecting the popular and the unpopular kids together, once and forever! Her friends list might be several miles long, but does it come from a genuine desire to befriend, or is it just a means to mask her insecurities?

5.3 Stories and Levels

In *Prom Week*, the player has a choice between two game modes: story and freeplay. Story mode allows the players to both experience and create the narrative of a specific character's prom experiences. Each story is centered on one character and consists of three to five levels where the last level is always Friday night at the big dance – prom itself! Each level of the story takes place on successive days leading up to the prom. Levels are set at locations in the community and high school where the lead character of the story and other (strategically placed) members of the cast are present. The player progresses through levels and accrues a social state and social history according to their gameplay choices until they finish the final level set at the prom. When they reach the ending of the last level, the player is rewarded with one of several longer capstone performances that is tailored to their gameplay and the lead character of the story.

Figure 20 – The story goal interface in this screenshot is showing the goals for Naomi's story. It is available to the player during story mode via the "Goals" button to the bottom left. Each goal, such as "Prom Queen Dream", can be clicked on to show the social state that the player needs to achieve to satisfy the goal. Complete story goals determine how the end of a character's prom story unfolds.

5.3.1 Story Goals

To tell stories and help the player in fulfilling the story character's prom ambitions, stories are associated with goal social states for the player to reach. These story goals are specific to a character and set up a challenging set of obstacles that, if surmounted, will result in different endings to the story. The potency of the ending is reflected by the number of story goals the player has completed – if no story goals are completed, a default ending where the lead character fails will be performed. More goals completed results in a more dramatic ending that reflects the social space created by the player on the lead character's behalf. As an example, here are Simon's story goals (as seen in the screenshot in Figure 20):

- Totally Popular
  - Simon is friends with at least 5 people.
- L0V3 in the air
  - Simon is dating someone.
- Caught Cheating
  - Simon is dating at least 2 people.
  - Simon is cheating on someone.
- An Ideal Rival
  - Simon is friends with someone.
  - Simon and someone are enemies.
  - They are the same "someone."
- Pitiful me
  - At least 2 generally negative things happened to Simon.

The highest level items in the list are a description or name of the goal. The next level contains the English language description of what social state needs to be met to accomplish that goal. Each goal consists of a *CiF* condition and is a good example of how the AI system was used outside of its standard social model capability to enhance *Prom Week*. The English description of the story goal condition is procedurally generated from the list of predicates of the rule and of the type of character variables in the rule.

Figure 21 – This is a screenshot of the freeplay mode setup interface. A number of characters can be selected to be in the level. The maximum number of characters that can be selected is determined by the size of the level. Near the bottom center is a visualization of the social world (called the "social minimap" by the development team) that assists the player in understanding the social situation they are creating. New freeplay mode gameplay sessions can be started (with the "New Game" button) and old sessions can be continued (with the "Continue" button that is disabled in this screenshot).

5.4 Freeplay Mode

Without the story-related mechanics of levels, goals, and endings, freeplay mode is closer to a straight simulation than story mode. The player can choose a setting and create a cast of characters to be in the level (of a maximum size dependent on the level's size). This mode was intended to be a test bed for tuning the social world and to provide a different game play mode for the player that is more open to exploration and experimentation of the interactions between characters. The freeplay mode menu can be seen in Figure 21.

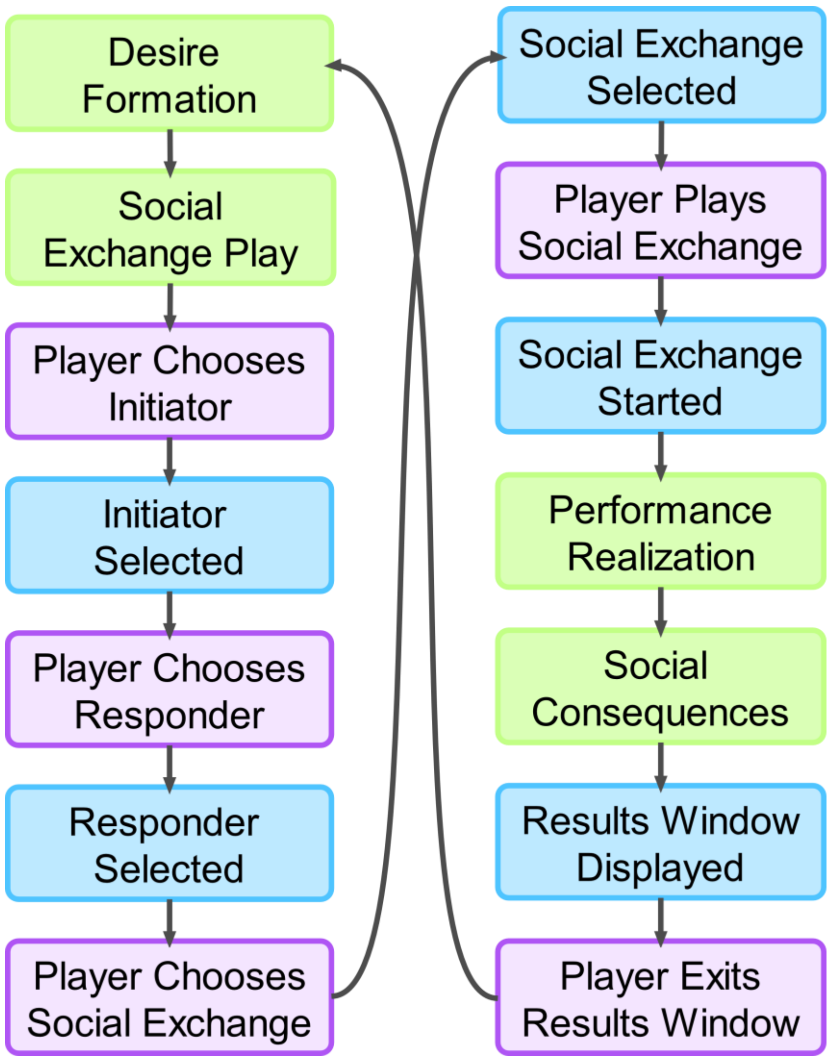


Figure 22 – This figure shows the flow of interaction between the player, *Prom Week*, and *CiF*. Player Chooses Initiator, Player Chooses Responder, Player Chooses Social Exchange, Player Plays Social Exchange, and Player Exits Window are player actions. Initiator Selected, Responder Selected, Social Exchange Selected, Social Exchange Started, and Results Window Displayed are *Prom Week* interface states. Desire Formation, Social Exchange Play, Performance Realization, and Social Consequences are *CiF* processes.

6 CiF and *Prom Week*

This section consists of detailed descriptions of how parts of *CiF* were integrated with *Prom Week*. Because *Prom Week* has gameplay needs outside of *CiF's* authoring strategy, *CiF's* data structures and processes were used outside of the standard authoring flow. How and why this out-of-band integration was put to use is discussed in the last half of this section. This section describes what was authored for *Prom Week*; the how and why is covered in the Authoring for *Prom Week* section of chapter 7. The basic flow of interaction between the player, *Prom Week*, and *CiF* is shown in Figure 22.

6.1 Relationships, Statuses, and Traits

Prom Week has three relationships (dating, friends, and enemies) and 34 statuses; 21 of them are statuses associated only with the character that has the status and 13 are directed statuses linked to another character. Statuses have default durations of 3 turns until they are removed from the character. Many of the statuses, such as popular and cheating, are continually renewed by triggers.

Each character has between 3 and 5 of the 49 traits. Their assigned traits give each character a unique feel during performances as they are linked in how characters perform. Traits are featured in trait-specific microtheories, non-trait specific microtheories with trait special cases, and performance conditions. By being considered at these various levels in *CiF*'s process, traits have a heavy impact on the ways characters interact. A complete list of statuses and traits can be found in Table 11. For more insight into how and why these specific relationships, statuses, and traits were authored for *Prom Week,* see the Authoring for *Prom Week* section in chapter 7.

Table 11- The traits and statuses in Prom Week. The two types of statuses, directed and not directed, are broken up into separate columns.

Traits		Statuses	
		Not Directed	Directed
Outgoing	Shy	Embarrassed	Homewrecker
Attention Hog	Impulsive	Cheater	Has A Crush On
Cold	Kind	Shaken	Angry At
Irritable	Loyal	Desperate	Wants To Pick On
Loving	Sympathetic	Class Clown	Annoyed With
Mean	Clumsy	Bully	Scared Of
Confident	Insecure	Love Struck	Pities
Mopey	Brainy	Grossed Out	Envies
Dumb	Deep	Excited	Grateful Toward
Shallow	Smooth Talker	Popular	Trusts
Inarticulate	Sex Magnet	Sad	Feels Superior To
Afraid Of Commitment	Takes Things Slowly	Anxious	Cheating On
Domineering	Humble	Honor Roll	Cheated On By
Arrogant	Defensive	Looking For Trouble	Homewrecked
Hothead	Pacifist	Guilty	
Ripped	Weakling	Feels Out Of Place	
Forgiving	Emotional	Heartbroken	
Swinger	Jealous	Cheerful	
Witty	Self Destructive	Confused	
Oblivious	Vengeful	Lonely	
Competitive	Stubborn	Homewrecker	
Dishonest	Honest		
Male	Female		
Wears A Hat	Muscular		
Cares About Fashion			

6.2 Microtheories

Prom Week's microtheories provide the general social knowledge (they are not linked to any particular social exchange and can potentially be used by all exchanges) for characters to form desires over twelve intents of social exchanges: six from raising and lowering each network and six more for starting and ending relationships. Each microtheory captures a small domain of high school social reasoning. The domains can be as straight forward as “excited” or as complicated as "dealing with someone who was mean to my friend” or “dealing with someone who did something romantic to you and you don’t feel that way about them”.

There are total of 145 microtheories containing a grand total of 3044 influence rules. Figure 23 shows the possible weights per intent that microtheories can contribute to a character’s desire to play social exchanges. Influence rules in the microtheories typically have a weight between -5 and 5 and can go up to a magnitude of 15 when encoding exceptional social circumstances.

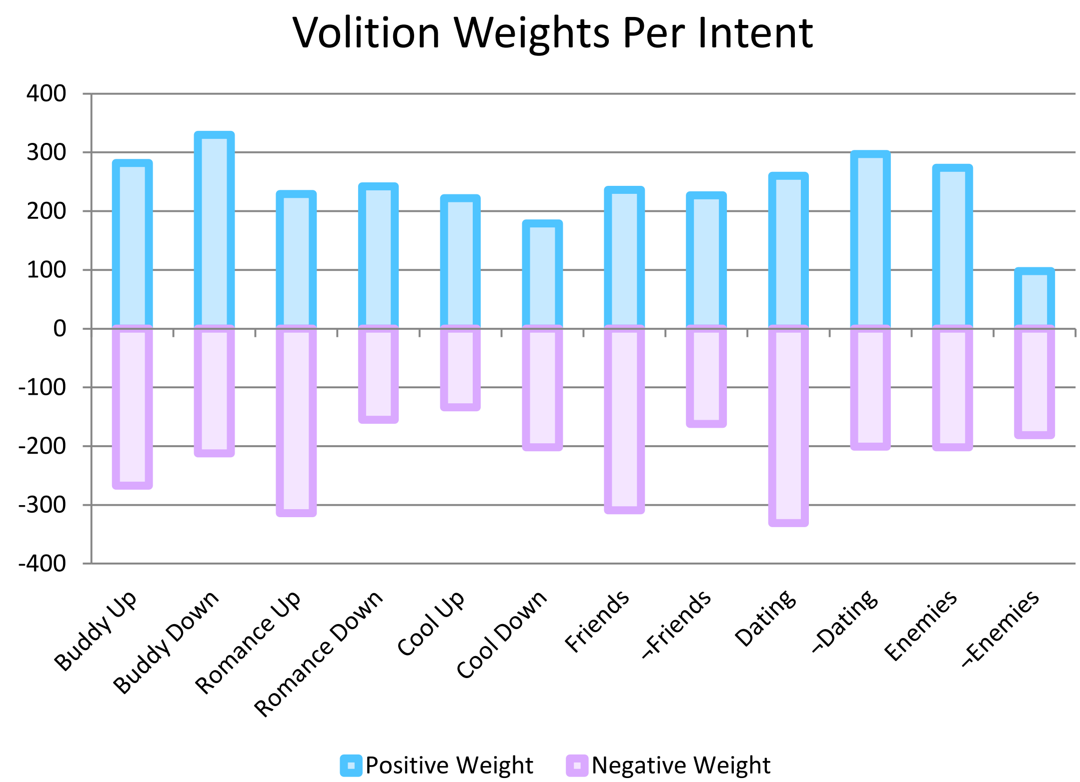


Figure 23 – The total weights possible from microtheories for each intent type.

Not all microtheories are equally important to gameplay; not all bits of social knowledge have the same impact on the social behaviors of high school students. As is discussed in chapter 7's Microtheories section, microtheories were authored in an importance order: relationships, social network values, statuses, and traits. As shown in Table 12, the three microtheories that contain more than 100 influence rules each are Dating, Enemies, and Friends. They are followed by a few microtheories over social networks (Cool Low, Cool High, BudNet Low, RomNet High) intermixed at the lower end of the influence rule count by those of the most impactful statuses (Guilty, Heart Broken, Lonely, Angry At). These are followed by microtheories about traits (Sex Magnet, Afraid of Commitment, Attention Hog) and then microtheories with compound definitions ("Dealing with someone who did something negative to you", "Dealing with someone who is cheating on you", "Someone macking on your crush").

Table 12 – This table shows the number of influence rules in each of Prom Week's Microtheories. The entries with no rules were either intended to be authored or were removed for game balance.

Influence Rules in *Prom Week's* Microtheories			
Dating	147	Arrogant	12
Enemies	123	Brainy	12
Friends	122	Cold	12
Dealing with Bully	82	Confident	12
Cool Low	76	Defensive	12
Cool High	69	Domineering	12
BudNet Low	67	Emotional	12
Guilty	65	Hothead	12
Heart Broken	63	Impulsive	12
Lonely	59	Insecure	12
RomNet High	58	Jealous	12
Angry at	57	Loving	12
Confused	57	Mean	12
Shaken	56	Oblivious	12
BudNet High	51	Pacifist	12
Love Struck	47	Self Destructive	12
Embarrassed	45	Shallow	12
Pities	45	Smooth Talker	12
Dealing with a Cheater	45	Swinger	12
Anxious	44	Takes Things Slowly	12
Envies	44	Weakling	12
Wants To Pick On	43	Friend's BudNet Avg Low	12
Dealing with a Popular person	43	Friend's CoolNet Avg Low	12
Sad	39	Dealing with someone who did something negative to you	12
Desperate	38	Dealing with someone who did something positive to you	12
Annoyed With	36	Something negative just happened to you	12
Dealing with Class Clown	35	Something positive just happened to you	12
Bully	33	Dealing with a Dating Taken	12
RomNet Medium	29	Dealing with a Cheating On you	12
Has a crush on	29	Dealing with someone who just did something mean to you	12
Excited	29	Someone macking on your S.O.	12
Grateful Toward	29	Dealing with someone who was mean to my friend	12
Scared of	28	Someone macking on your crush	12
Dealing with Honor Roll	28	Dealing with someone who used to cheat on you	12
Cheerful	27	Dealing with your S.O. who has been macking on others	12
Looking for Trouble	26	Dealing with someone who just had something bad happen to them	12
BudNet Medium	24	Dealing with someone that a bunch of good things happened to	12
RomNet Low	23	Dealing with someone who did something romantic to you and you don't feel the same	12
Feels out of place	22	Dealing with someone who used to be cheating on you	12
Dealing with a Sex Magnet	22	Dealing with someone who used to be homewrecking you	12

Trusts	19	Mean to your sweetie	12
Cheater	17	Misunderstanding	12
Popular	17	Misunderstanding A LOT	12
Class Clown	16	Dating BudNet Avg High	11
Grossed out	16	Cheated on By	11
Cool Medium	14	Homewrecked	11
Sex Magnet	14	Friend's CoolNet Avg High	10
Dating Taken	14	Dating BudNet Avg Low	7
Friend's BudNet Avg High	13	Dating RomNet Avg High	7
Cheating On	12	Dating RomNet Avg Med	6
Being Cheated On	12	Dating CoolNet Avg Med	6
Afraid of commitment	12	Friend's CoolNet Avg Med	6
Attention Hog	12	Dating CoolNet Avg High	5
Clumsy	12	Knight in shining armor	5
Deep	12	Friend's RomNet Avg Low	4
Dishonest	12	Friend's RomNet Avg Med	4
Dumb	12	Friend's RomNet Avg High	4
Forgiving	12	Failed romance makes you want romance down	4
Honest	12	Enemy BudNet Avg High	2
Humble	12	Dating BudNet Avg Med	1
Inarticulate	12	Dating CoolNet Avg Low	1
Irritable	12	Enemy BudNet Avg Low	1
Kind	12	Friend's BudNet Avg Med	1
Loyal	12	Dating RomNet Avg Low	0
Mopey	12	Enemy BudNet Avg Med	0
Outgoing	12	Enemy RomNet Avg Low	0
Ripped	12	Enemy RomNet Avg Med	0
Shy	12	Enemy RomNet Avg High	0
Stubborn	12	Enemy CoolNet Avg Low	0
Sympathetic	12	Enemy CoolNet Avg Med	0
Vengeful	12	Enemy CoolNet Avg High	0
Witty	12	Dealing with someone who was mean to my date	0

6.3 Social Exchanges

Because they are the only way a player can affect the storyworld of *Prom Week,* social exchanges are extremely important and required the most development and authoring effort out of all of the parts of the game. They were required to be flexible enough to fit many different social contexts while being specific enough to provide interesting performances. To draw a comparison to Crawford's interaction loop (Crawford 2002), which was previously mentioned in chapter 1, forming character desires about and performing social exchanges occupies the critical space in *Prom Week*'s design that connects what the player "speaks" to how the system responds by "thinking" and "speaking" in return.

There were a grand total of 39 social exchanges in the released version of *Prom Week*. To avoid repetitious performances and to better match performances to social context, each social exchange has many instantiations each of which has several lines of dialogue. The exact numbers of instantiations and lines of dialogue for all social exchanges can be seen in Table 13. *Prom Week* features 818 instantiations and 4987 lines of dialogue. As a complement to the social change of a social exchange performance, every line of dialogue can be tagged with animations that indicate part of the social change resulting from the interaction (such as a heart breaking for when dating is stopped or a blue arrow pointing up with character's face next to it to indicate an increase in a cool network value).

Table 13 – Social exchanges, their intents, number of instantiations, and the number of lines of dialogue are shown in this table.

Name	Intent	Instantiations	Lines of Dialogue
Annoy	buddy down	19	105
Ask Out	dating	43	311
Back Me Up!	buddy up	31	214
Backstab	enemies	14	102
Bicker	buddy down	25	142
Blow Off Plans	end friends	16	95
Brag	cool up	41	230
Brash Remark	romance down	15	86
Broke My Heart!	end dating	9	53
Brutal Break-Up	end dating	13	112
Bully	enemies	25	134
Concede	end enemies	16	107
Confide In	friends	27	207
Declare War	enemies	13	93
Embarrass Self	cool down	16	79
Flirt	romance up	28	164
Give Advice	buddy up	40	280

Humble Self	cool down	17	121
Idolize	cool up	26	136
Insult	buddy down	28	131
Insult Friend of	enemies	22	122
It's Over!	end dating	13	86
Leave Me Alone!	end friends	14	87
Let's Be Friends!	end dating	12	76
Make Plans	friends	24	149
Make Up	end enemies	15	83
Open Up	romance up	12	55
Physical Flirt	romance up	18	99
Pick-Up Line	dating	23	134
Reminisce	buddy up	28	149
Share Interest	buddy up	26	137
Shoot Down	romance down	22	148
Show Off	cool up	21	98
Spread Rumors	buddy up	10	94
Turn Off	romance down	19	100
Txt Break Up	end dating	17	111
Weird Out	cool down	19	110
Woo	romance up	28	145

6.4 Game Play Integration

As *CiF* is meant to be a playable model of social interaction and not a complete game system, the development of *Prom Week* required that *CiF* be used in ways unique to *Prom Week*. This section describes the ways that *Prom Week* leverages the knowledge representation and processes of *CiF* in service of gameplay.

6.4.1 Story Goals

Stories in *Prom Week* are a series of simulation environments from which the player is expected to make author-specified social states, known as story goals, true in the world. The fiction of the storyworld is used to provide reasons for the players to interact with the simulation in order to achieve those social states. Basically, *Prom Week* is a game in which players solve social puzzles by manipulating the characters in ways that achieve the goal social states. This provides an end state for the player to experience and facilitates the character in the construction of a narrative that transforms the story world from its initial state to fulfilling story goals. The difficulty of achieving the story goals in a level defines the challenge of that level.

Each story has at least three and at most five story goals. Story goals are composed of conditions that are evaluated after each round of gameplay. If progress is made or undone by playing a social exchange, the player is told so via the notifications located at the bottom right of the UI.

6.4.2 Story Endings

Story endings are special cases of social exchange instantiations. They are customized, procedurally-chosen conclusions to stories based on the choices made by the player. As they end the player's journey through a story, they have no social change. However, their conditions are tightly bound with story goals as they serve to culminate the player's path to achieve story goals. If the player achieves no story goals, a generic (and often sad or unsatisfying to the character) story ending is performed. Story endings commonly resolve the story created by a single story goal. However, some of the more dramatic endings are authored to bring more than one completed story goal to conclusion. For example, Doug's story has four story goals. However, as is shown in Table 14, it has eight endings with four that relate to a single goal, one related to no goals, and three related to two goals each. Scripts of endings addressing story goals are in chapter 7's Story Endings section.

Table 14 - The relationships between the story goals and story endings in Doug's story. The story endings are named in the first column and the story goals are the column headers. Check marks indicate the story goals concluded by the corresponding story ending.

Story Goal	Big Buddy Doug	Super Nice Guy	Kinda Mean Dude	Last Minute Date
Kickin' It				
Words of Wisdom	✔			
Little Kindness, Long Way		✔		
Burning Bridges			✔	
Shining Armor				✔
All Around Cool Kid	✔	✔		
Bad Boys Get All the Babes			✔	✔
Something Special	✔			✔

The initiator is always bound to the story lead character in story goals while the responder and possible other can be any of the characters in the level. As a player can see more than one ending, *Prom Week* avoids having the same character as the responder in multiple endings. In the case where the story goals Big Buddy Doug and Kinda Mean Dude are both satisfied, the player is presented with both Words of Wisdom and Burning Bridges. Because it would be awkward for the same character to both receive sagely advice and an angst-laden retribution in succession, we always pick unique responders for each ending unless it is completely unavoidable.

6.4.3 Story Points

As stated in the Basic Flow of Interaction section, story points allow the player to bend the simulation to their will. This mechanic was added to *Prom Week* to give the player some way to handle particularly hard social situations that occur as a byproduct of simulation. Practically, it provided an avenue for tuning difficulty at the cost of players being able to take shortcuts or bring about situations that are normally outside of social norms.

The player starts a story with 50 story points and gets 10 per turn of gameplay. Table 15 shows the cost for each way story points can be spent. The cost of changing the outcome of a social exchange is based on the strength of the volition of the responder (weak, medium, and strong) and the intended change type of the exchange (network or relationship). Unlocking social exchanges has a cost ($cost$) based on difference between highest volition of any social exchange (v_m) from the initiator to the responder and the volition of the exchange to unlock (v_u): $cost = 2\ (v_m - v_u)$. A minimum value of 10 and maximum value of 100 is enforced on the cost.

Table 15- The cost associated with affordances purchasable with story points. Changing the outcome of a social exchange (for both network and relationship types) has a difference cost based on weak, medium, and strong reactions from the responder (respective seen in the cost of those games). The cost for unlocking a social exchange is based on its relative volition to the highest volition exchange.

Story Point Affordance	Cost
See Responder's Motives	6
See Outcome	15
Change Outcome of Relationship Exchange	20/25/35
Change Outcome of Network Exchange	15/20/25
Unlock Exchange	10-100

6.4.4 Stars

With the social minimap and the top five social exchanges available to the player, they have both a broad and a fine view of the desires of characters. During playtesting, we found that these two views can be difficult to combine in a way that helps the player understand the social life of an individual character. To this end, we added a view that would process a character's desires into a middle-of-the-road view by using the prospective memory and some hard coded conditional logic. Called "stars", this interface represents other characters important to the selected character in distinct star categories. If no character fits a star role, a silhouette is shown. The following is a list of the star roles and how characters are chosen to fill them (in the examples, x is bound to the selected character whereas y is a free variable):

Best Friend. The best friend is the other character who has the highest buddy social network score from the selected character and satisfies the following condition:

relationship(Friend,x,y) and network(Buddy,x,y) > 66

Dating. This role is similar to best friend but chooses from dating relationships and romance network values:

relationship(Dating,x,y) and network(Romance,x,y) > 66

True Love. In *Prom Week*, it is common for a character to be dating one character while having strong romantic feelings for another. This role captures this by displaying the character with the highest romance network value in the eyes of the selected character that makes the following condition true:

status(HasCrushOn,x,y) and network(Romance,x,y) > 66

Idol. The coolest character (via cool network value) in the eyes of the selected character with a minimum cool network value of 66 is bound to this role.

Lamest. The character with the lowest cool network value from the selected character is bound to the lamest star role if they satisfy the following condition:

¬relationship(Friend,x,y) and network(Cool,x,y) < 20

Rival. The character with the lowest buddy network value from the selected character that makes the following condition true takes the rival role:

relationship(Enemies,x,y) and network(Buddy,x,y) < 34

6.4.5 Social Status Updates

Social status updates were included in the game to make the characters feel more alive and to promote the idea of a living, breathing social world. They are the *Prom Week* fictional world equivalent of posts or status updates in social media; they are not intrinsically related to statuses in *CiF*. There are 226 potential social status updates each of which has an associated social condition and an NLG template. The social status updates are processed while exchanges are performed. Their associated conditions are evaluated and, if true, scored for saliency (just like the performance conditions for social exchanges). Periodically, a social status update is selected via weighted random selection based on the salience score and displayed.

Table 16 – This table contains examples of social status updates in Prom Week. The condition column contains the predicates that must be true for its associated NLG template (in the right column) to be potentially posted to the notification area.

Condition	NLG Template
network(Buddy,i,r) > 70	*@%r% %gender(r,dude,girl)%, real talk: i think our friendship transcends time and space, yo*
network(Buddy,i,r) > 50 and network(Buddy,r,o) > 50 and network(Romance,i,o) > 50	*@%r% %gender(r,dude,girl)%, can you pleeeeaaase talk to %o% for me? i will owe you one big time*
¬relationship(Friends,i,r) and network(Buddy,i,r) < 50 and SFDB(Nice,y,x,window:1)	*%random(maybe,could it be,I guess)% %r% isn't such a %pejorative% after all...*
trait(Deep,x)	i could spend hours staring at the night sky, such beautiful mysteries... @_@
¬relationship(Dating,x,y) and relationship(Dating,y,z) and network(Romance,i,r) > 65 and SFDB(Flirt,i,r,window:3)	O @%r%... deny thy %o% & be but sworn my love

The condition portion of social status updates is used to parse the social world for small social contexts. Establishing these social contexts provides a foundation for a character to comment on the storyworld. Social status updates tend to fall into three types of comments. The first are comments expressing the individuality of a character. In Table 16, an example of this type can be seen in the row that contains a condition of trait(Deep,x). A second type is a character commenting on their general social situation (seen in the network(Buddy,x,y) > 70 example). The final type consists of reactive comments on the recent changes in the social world. These comments rely on the window property of SFDB predicates and of predicates of type True in History. By having a condition that contains a predicate with a small window like SFDB(Nice,y,x,window:1), characters can react to the outcome of social exchange play without being directly involved.

6.4.6 Motives and Factors

Given the complicated social space developed for *Prom Week*, players often have a difficult time understanding how and why characters make the decisions they do. The complete set of information (all of the influence rules that contributed to a decision) would only serve to overwhelm the player in detail. We addressed these issues by showing the player a slice of the reasoning based on predicates known as motives and factors.

The motives of the initiator to start the social exchange and the factors of the responder to either accept or reject the game are based on the influence rules that contribute weight to the decision. Each predicate in every true influence rule is given a factor weight based on the weight of the influence rule divided by the number of predicates in that rule. Each of these newly-weighted predicates is sorted. Weights of matching predicates (with matching criteria of predicate type, initiator, and responder) are summed. The predicates with the highest summed weight are presented to the player as motives and factors.

The following example shows how motives, factors, and their natural language descriptions are created. In the context of Zack (the initiator) playing Ask Out with Lil (the responder), the following influence rules were responsible for Zack's desire to Date lil:

network(Buddy,i,r, >66) and status(HasCrushOn,i,r) → intent(relationship(Dating,i,r))+2

status(HasCrushOn,i,r) and status(HasCrushOn,r,i) → intent(relationship(Dating,I,r))+3

The first influence rule increases the desire of an initiator to date a responder if they are good friends and the initiator has a crush on the responder. The second influence rule increases the desire of an initiator to date a responder if they have mutual crushes on each other. Each of the predicates in the conditions of the influence rules are associated with a number that represents that predicate's contribution to the intent. This number is equal to the weight of the intent divided by the number of predicates in the condition of the influence rule. In this case, the predicates and contribution values of the condition of the first rule would are a value of 1 network(Buddy,i,r, >66) for and 1 for (HasCrushOn,i,r). The second rule's predicates would have values of 1.5 (HasCrushOn,i,r) for and for

(HasCrushOn,r,i). Identical predicates are merged into one entry with all of their contribution values summed. This would result in the (HasCrushOn,i,r) predicate found in both rules having a contribution value of 2.5. This merged list of predicates is then sorted in descending order of contribution value. Up to the 10 entries in this list that have the greatest contribution values are turned into natural language via a process similar to filling out performance NLG templates but customized for short phrases. Table 17 shows the contribution value list and the natural language translation of its entries:

Table 17- This table shows an example of a list of predicates and contribution values used in determining the character motives and factors in *Prom Week*. The third column shows the results of entries being translated into natural language.

Predicate	Contribution Value	Natural Language Translation
status(HasCrushOn,i,r)	2.5	*Zack has a crush on Lil.*
status(HasCrushOn,r,i)	1.5	*Lil has a crush on Zack.*
network(Buddy,i,r,>66)	1.0	*Zack thins Lil is a good buddy.*

Chapter 5

Evaluation of *Prom Week*

This chapter focuses on evaluating the research contributions (described in chapter 1) through an analysis of *Prom Week*. The first section describes the main source of data for this analysis, namely play traces gathered from the players of *Prom Week.* The second section evaluates the first research contribution, summarized as enabling playable social interaction in fictional worlds, in two parts: how the gameplay was procedurally fitted to players' actions and how players could guide the gameplay in strategic ways to achieve story goals. Allowing the authoring of a new level of complexity in player choice, the second contribution will be evaluated by comparing the complexities of storyworlds in *Prom Week* and CRPGs with the most complex story content.

1 Playable Social Interaction in Fictional Domains

To evaluate *Prom Week* and *CiF* in light of system responsiveness and variation to player actions, play traces can be analyzed to determine how *CiF* is responding to real play situations. Even with the large amount of variation supported by *CiF* in a storyworld as content rich as *Prom Week*, there are reasons why players could potentially be exploring a very small space of the possible story. The cast of characters in a level could have very little desire to interact with one another. Overly restrictive story goals could be constraining player choice into narrow spaces of interaction. The balance of microtheories and applicable social exchanges could leave few social exchanges for the player to choose from. Even with involving players from *Prom Week*'s alpha to its release, only a small slice of the possible game states could be seen from user testing.

To gain a better understanding of the variation in stories that players experience in the wilds of public release, a holistic and detailed understanding of the play traces is useful.

1.1 Play Traces from Prom Week

As players interact with *Prom Week*, the system saves their interactions with the game. These traces provide data for saving and continuing play sessions and contain the information needed to re-simulate the social state created by the player. After the player exits a play session or completes a level, *Prom Week* sends a trace to a server. The trace is associated with an anonymized ID that represents the player and is used to track a player across play sessions.

Each play trace consists of the game events chosen by the player that have an effect on the social world. Each event is stored with enough context to recreate the social world constructed by the player given *Prom Week*'s initial state. The SFDB was designed to keep a record of *CiF*'s activities and the social exchanges played, statuses timed out, and triggers fired. Additionally, *Prom Week* uses the SFDB to store when and how the player uses story points. When sent to the server, the SFDB is made into XML with included data about the level. Here is an example play trace:

```xml
1:  <LevelTraces startTime="0" endTime="3" name="Thursday: After Class"
    storyName="Zack" juice="80" type="endOfLevel">
2:    <SFDB>
3:      <StatusContext time="2" status="pities" from="Naomi" negated="true" to="Simon"/>
4:      <StatusContext time="2" status="wants to pick on" from="Zack" negated="true"
    to="Lil"/>
5:      <StatusContext time="2" status="pities" from="Monica" negated="true" to="Simon"/>
6:      <TriggerContext time="2" id="1" initiator="Zack" responder=" "/>
7:      <SocialGameContext gameName="Confide In" initiator="Naomi" responder="Zack"
    initiatorScore="-5" responderScore="7" time="2" effectID="23" other="Monica">
8:        <SFDBLabel type="mean" from="Naomi" to="Monica"/>
9:      </SocialGameContext>
10:     <JuiceContext type="1" cost="34" order="3" time="2" gameName="Confide
    In" initiator="Naomi" responder="Zack"/>
11:     <SocialGameContext gameName="Insult Friend of" initiator="Zack"
    responder="Monica" initiatorScore="-2" responderScore="6" time="1" effectID="16">
12:       <SFDBLabel type="mean" from="Zack" to="Monica"/>
13:     </SocialGameContext>
14:     <SocialGameContext gameName="Make Plans" initiator="Zack" responder="Naomi"
    initiatorScore="9" responderScore="-9" time="0" effectID="0"/>
15:   </SFDB>
16: </LevelTraces>
```

Line 1 contains data about the level of the trace including when the level was started (in *CiF* turns), when the level was ended (also in *CiF* turns), the name of the level, the lead story character, the story points (which are referred to as their development name "juice" in the level traces), and the type of play trace. The type can be freeplay, end of level, end of story, or user exit. Line 2 is the start of the SFDB data.

Each SFDB entry is a separate XML tag child of the SFDB tag parent. Every entry is tagged with a type (designated in the name of the tag) and a time. Lines 3-14 show examples of all 4 types of entries. Line 3 shows a status entry in which Naomi loses the

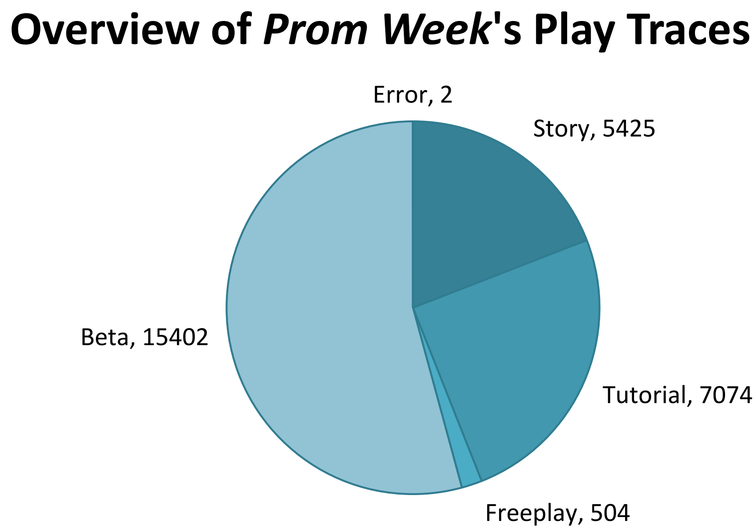


Figure 24 – An overview of the player traces generated by *Prom Week's* players from on December 10th, 2011 to May 17th 2012.

status of pitying Simon. A trigger with an id value of 1 (if 4 or more people think you are cool, you become popular) is fired on Zack in line 6. Line 7 is an entry over the "Confide In" social exchange that was played between Naomi and Simon. This entry contains the information needed to recreate the performance: the initiator, responder, the specific performance (effectID), volition of the initiator (initiatorScore), responder's choice (responderScore), and the SFDB label placed on the performance (in the form of the SFDBLabel child tag). An example of a story point entry is on line 10 as a JuiceContext tag type. In this case, the id of 1 means the story points were spent to reveal the outcome of the "Confide In" exchange before it was played (at the cost of 34 story points).

From when play traces were first collected in beta on December 10th, 2011 to May 17th, 2012, players have generated a total of 28,407 traces. Of these traces, 7,074 took place in tutorial levels, 504 were of the goal-less freeplay mode, and the remaining 5425 took place in *Prom Week*'s stories (see Figure 24). Only the 5425 story play traces generated after the official release of *Prom Week* on February 14th 2012 are used in this evaluation.

The story play traces were generated each time a level was successfully ended (either the level clock was clicked or the player ran out of time) or a story ending was reached (a prom ending was seen). The release version of *Prom Week* had 5 playable stories: Doug, Oswald, Simon, Monica, Edward and Lil (for a small time right after release, Naomi's story was playable).

1.2 Gameplay Customized Storyworld Exploration

To get a sense of how *CiF*'s simulation and *Prom Week*'s gameplay impact the actual choices presented to the player, level traces were analyzed and visualized using the Façade Log Analysis and Visualization Tool (Sali and Mateas 2011; Sali 2012), a visualization tool that aims to enhance the current toolset for studying interactive narratives. This tool helped in forming an understanding of how players were interacting with the released version of *Prom Week*. Even though the player has many options of social exchanges to choose from, it is not clear without evaluation that there are enough paths through the story space to satisfy the whims of each individual player. Furthermore, story goals, level casts, and the desires of the characters themselves may restrict the options available in

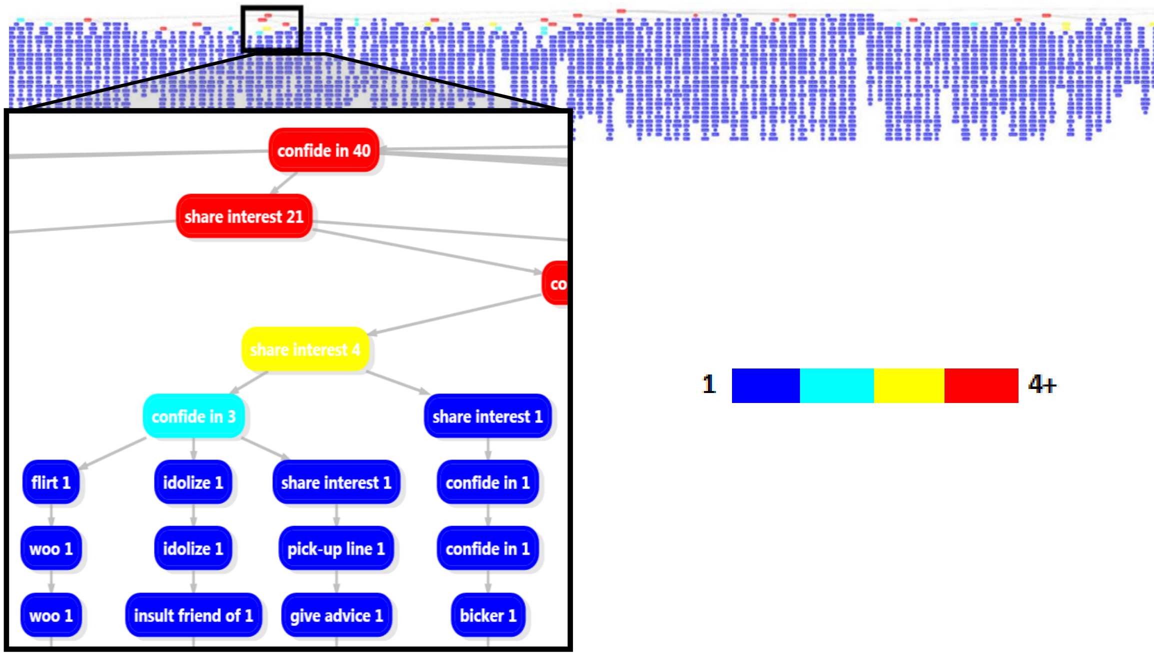


Figure 25 - A play trace graph showing how often each distinct path through Simon's story was taken (shown by the color and number associated with each node). The large band of nodes seen at the top of the diagram represents approximately one third of the total size of the complete map. The cutout shows a section of the map in detail including examples of social exchanges (like "pick-up line" and "confide in") that appeared in more than one play trace. The majority of play traces are unique.

such a way that many players will be forced down a narrow few paths in their pursuit of story goals.

We were pleased to discover that there was a very large degree of variation in the way that players navigated the social space. Examining a tree map representing the social moves selected during the final level of Simon's campaign reveals that, of the 263 unique playthroughs we analyzed, no two were exactly alike; the space was rich enough to allow for an entirely unique play trace per player. Figure 25 is a tree graph of the play traces analyzed for Simon's campaign. Each node represents a selected social exchange, each of which results in changes to the game state (e.g. relationships starting or ending). A path through the tree is the sequence of social exchanges a player made from the starting state in the first level (the root), to an ending (a leaf). Although there are a fixed amount of maximum turns in Simon's campaign, not all paths in the tree are the same length as players have the option of skipping remaining turns and jumping ahead to the next level. The color of the nodes is a heat map indicating frequency of node visitation along that specific path; red is frequently visited (i.e. several players followed that exact same route up to the point of that node), and dark blue means visited only once (i.e. the route to that node was experienced by only a single player). For readability purposes, the nodes have been collapsed to the names of social exchanges selected, when in actuality gameplay moves are identified by the social exchange and the two characters to perform that social exchange. Including this differentiator would have further increased the branching of the tree, but we claim that it is already branchy enough for the purposes of validating our hypothesis of high variability.

The average indegree (times a node was encountered by a player) of a node in this graph is approximately 1.11; though as mentioned above there are a few nodes towards the beginning that are selected many times––"share interest" and "confide in" are popular starting moves, happening 91 and 40 times respectively—the vast majority of story traces have nodes that are visited precisely once. This means the play trace is unique because no other trace is composed of the same sequence of social exchanges.

Performing n-gram analysis[29] revealed some interesting statistics on the patterns of sequences of social moves played (this analysis is explore in more detail in the next section). Using 1-gram analysis, there are 38 unique social moves that players employed on this level, out of a total possible 39 social moves that exist in the game. Using 3-gram analysis, we have 2521 unique patterns, of which only 80 appear more than 10 times. With 6-gram analysis, there are 5066 unique patterns of social exchanges, one of which occurred 16 times, another 10 times, and all the rest less than 5 times. The fact that so many separate patterns exist, with so little repetition, indicates that players were able to find their own way through the story space. Moreover, the n-grams that have the most repetition are situations in which the same social exchange was played multiple times in a row. Though apparently there is a player type that relies on a strategy of brute force (for example, attempting to 'woo' six times in a row), they are dwarfed by the number of other patterns exhibited.

[29] N-gram analysis is used to find repeated patterns of varying lengths in corpora.

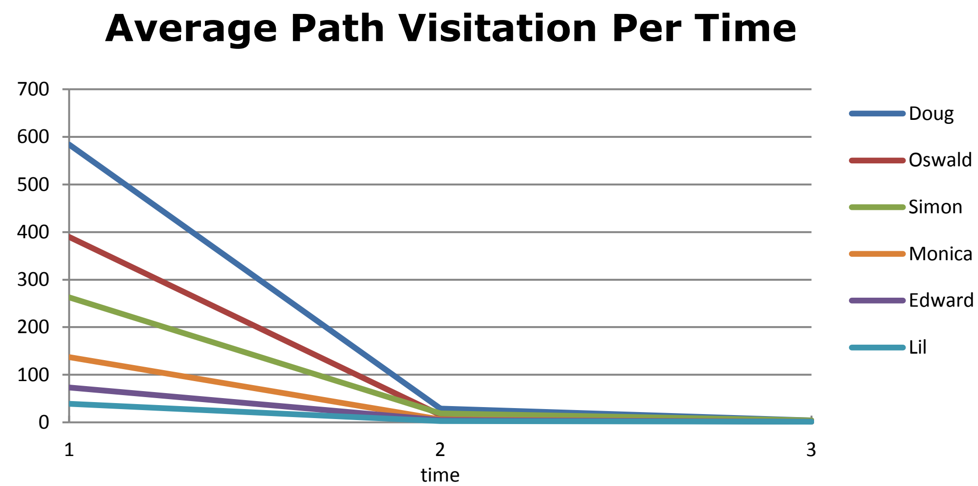


Figure 26 - This plot shows how unique each player's path through the story space is as time progresses. The x-axis is time, or number of turns, and the y-axis the average of how many times a story path has been visited.

Another interesting point was discovered by examining the tree graph of social exchanges. The sheer breadth of the tree gives a positive view of just how much variability there is in player choice; not only does the system allow for variability, but players are taking advantage of it as well. Additionally, though there are only 11 nodes that players chose for the first move, there are 79 different nodes selected for the second, and 143 for the third. By the fourth turn, nearly every gameplay trace is unique (see Figure 26). Even traces with subtle differences in gameplay actions (for example, the sequence of social actions "reminisce", "confide in", "ask out" as opposed to "confide in", "reminisce", ask out") can result in remarkably different traversals through the social state, as *Prom Week* keeps track of the specific social exchanges and instantiations that the user has seen and incorporates them into future social exchange selection. Moreover the specific ordering of social changes also impacts the formulation of which social exchanges characters want to play with each other, thus even seemingly similar play traces can be considered unique.

The general trend of paths becoming unique can be seen across the stories and is even more prevalent in the more difficult stories of the late game. Take Oswald's story as an example, which has 390 level traces that all begin in the same starting state. Twenty-five different opening moves were selected with an average indegree of 15.6. After the second turn the average drops to 2.36. The average dips to 1.27 after the third turn, and hits 1.07 after the fourth.

The above supports our first hypothesis of the variability in *Prom Week*. The low average indegree indicates that we are approaching a completely unique playthrough experience for each player; the large number of unique n-grams even for small n indicate that these unique playthroughs consist of different patterns of play; and the rapid branching factor means that the little overlap that does exist between players quickly separates into distinct traces. Given all of this, we claim that *Prom Week* was successful in providing a game space with large amounts of variability, even if, as we see below, players selected between only a handful of the total possible options on the first turn.

The relatively low variability seen during the first turn is actually positive evidence for our second hypothesis: that *Prom Week* is specifically providing large variability in the service of making stories playable. There are five characters in Simon's first level, and each character wants to engage in five possible social exchanges with each other character (the top five social exchanges character A wants to perform with B given the desires computed by *CiF* for character A). Since the player picks a unique initiator and responder, this means that there are at least 100 potential opening social exchanges (the actual number is a little higher, as players can spend story points to unlock additional options).

The fact that, of these hundred starting options, only eleven were ever pursued between all of the gameplay traces implies that players have been attempting to accomplish story goals. The beginning of each level provides framing text which contextualizes the characters' relationships to each other with relation to campaign goals, and offers small hints about how to accomplish the goals. The hints take the form of advising the player on which characters to form relationships with, but offer no advice on which specific social exchanges to try. This means that player actions are being motivated by story goals without being dictated by them, providing a solid foundation for our second hypothesis.

1.3 Strategy Driven Play

To determine if *Prom Week* promotes strategic play, this section analyzes the player-driven paths through *Prom Week* with respect to the successful completion of story goals. To be seen as an indicator for strategic play, large portion of the story paths - variable though they may be - need to lead to successful goals. Story goals in *Prom Week* represent story states for the player to make true in the storyworld. For example, in Simon's campaign, the player is tasked with accomplishing five distinct goals, including having Simon make five friends, having Simon begin dating someone, and giving Simon an "ideal rival" by making him friends and enemies with the same person. The combination of goals accomplished determines which ending for the campaign the player receives. Though endings are mostly pre-written to leverage authorial control, there still exists template dialogue within endings that allows for explicit references to specific social exchanges that were chosen by the player throughout the course of gameplay. This gives every choice the player makes—and not just goal completion—an impact on the campaign's climax.

1.3.1 Story Goal Completion

Figure 27 shows another view of the 263 traces which start at Simon's first level and progress their way through the end of his campaign. In this graph the color of the nodes shows the impact of that social exchange on story goals. Story goal completion ranges from dark blue to green, progress toward the goals is in the range of light blue to orange, and moving the social state away from the story goal (antiprogress) is colored orange or red. This data was generated by taking the same level traces used to generate Figure 27 and running them through *CiF*, keeping track of the goal accomplishments at each game turn.

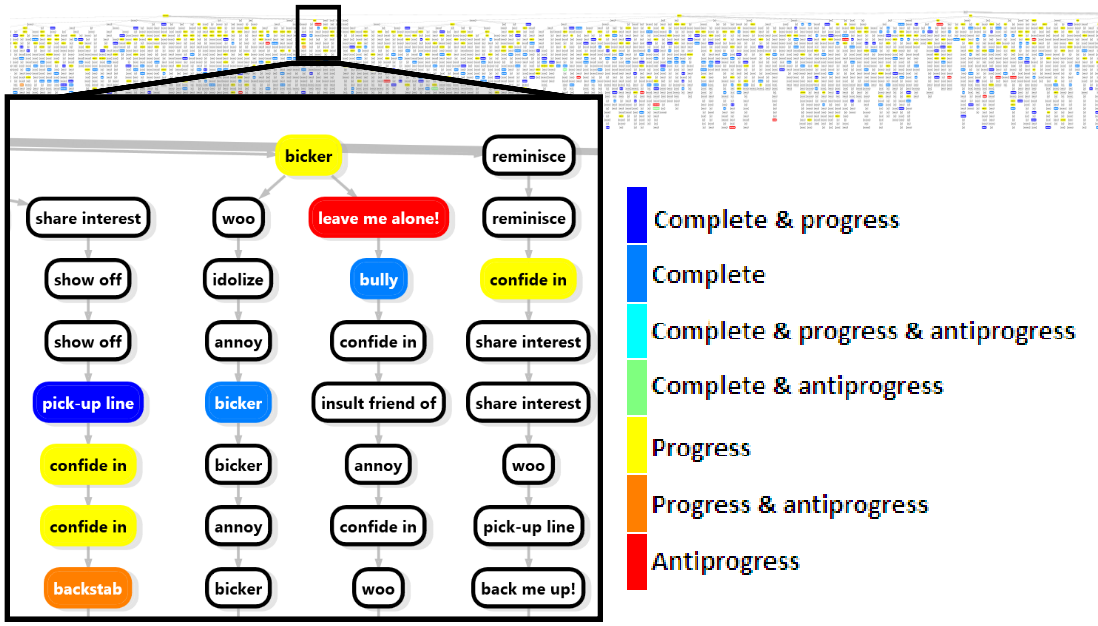


Figure 27 – A tree displaying the amount of progress towards goals in Simon's campaign. The color of the nodes represents the type of goal progress. There are three types of goal progress that can be combined in any way. Complete (Blue) means a goal was completed, progress (yellow) means that one aspect of a goal was made true, and antiprogress (red) means that an aspect of a goal that used to be true was made false. White nodes mean that no progress (or antiprogress) was directly made by making that social exchange, though the social state was still changed which could lead to progress in future turns. The large band of nodes along the top still represents about 1/3 of the total play traces of Simon's story.

Simon's campaign is the third non-tutorial level in *Prom Week* and is of intermediate difficulty. Though some goals can be accomplished in just a single turn (across all 263 traces for Simon's campaign, only 13 completed a goal on the first turn, and only 17 completed a goal on the second), the rest take several turns to complete. As seen in Figure 27, the story goals were completed by players at many points along the story paths. Of all of Simon's traces, only a single one did not contain any goal progress. All others exhibited at least some amount of effort towards achieving story goals.

Even though Simon's campaign is of intermediate difficulty, players still displayed an aptitude for achieving goals. Between all of the play traces, goal completion (on any of Simon's five goals) was reached a total of 610 times (average of 2.32 goals per player). If every trace from every file had accomplished all five goals, the total would be 1,315, which means that around 46% of all possible Simon goals were achieved. Goal progress was made a total of 837 times (average of 3.18 times per player), and goal antiprogress was made a total of 44 times (average of 0.18 times per player).

A concern when designing goals is that *Prom Week*'s gameplay—manipulating social relationships within a setting of cascading social influences in the pursuit of story goals—is fairly unique. Since *Prom Week* serves as an introduction to this genre of social puzzle game for most players, figuring out the nuances of the system to make story progress could have proven to be a challenge. Although the goal completion rate is perhaps a little low for a campaign of only intermediate difficulty, the results are encouraging because not only were players motivated to pursue story goals, they were also able to create a strong enough internal model of the storytelling system to be able to pursue story goals with some amount of success.

1.3.2 Common Play Patterns

Another way to see how players were strategically playing *Prom Week* is to look at their most common patterns of play and how those patterns related to what players want to accomplish. Primarily, we want to know the frequency of sequences of social exchanges played by the player and how they relate to the player's goals, which are typically the story goals.

Table 18 shows the most frequently used social exchange sequences from the play traces of Simon's story. Similar to an n-gram analysis, this table shows the 10 most common sequences of length 2 to length 6 with the number of times they were encountered. As previously mentioned, player-driven paths through stories quickly become unique as the length of the sequence increases. Similarly, the number of times each sequence was used decreases as its length increases. These sequences are abstracted from the actual play traces in that only the name of the social exchange is examined. This means that a Confide In between Simon and Zack would be the same as a Confide In between Naomi and Monica. However, a majority of the social exchanges in play traces involve the lead story character as an initiator or responder.

Table 18 – The common patterns of social exchanges played by players.

Count	1	2	3	4	5	6
378	confide in	confide in				
252	Woo	woo				
230	confide in	share interest				

183	Woo	confide in				
148	share interest	share interest				
142	give advice	confide in				
137	share interest	confide in				
128	confide in	give advice				
122	confide in	woo				
109	back me up!	confide in				
107	confide in	confide in	confide in			
99	woo	woo	woo			
69	confide in	confide in	share interest			
58	confide in	share interest	share interest			
53	woo	confide in	confide in			
52	woo	woo	confide in			
43	confide in	share interest	confide in			
41	share interest	share interest	confide in			
40	give advice	confide in	confide in			
36	confide in	woo	confide in			
47	woo	woo	woo	woo		
36	confide in	confide in	confide in	confide in		
24	confide in	confide in	confide in	share interest		
23	confide in	confide in	share interest	share interest		
22	confide in	share interest	share interest	confide in		
17	confide in	woo	confide in	confide in		
15	woo	confide in	confide in	share interest		
15	confide in	confide in	share interest	confide in		
15	confide in	share interest	confide in	share interest		
14	woo	woo	woo	confide in		
27	woo	woo	woo	woo	woo	
18	confide in	confide in	confide in	confide in	confide in	
8	confide in	confide in	confide in	share interest	share interest	
8	confide in	confide in	share interest	share interest	confide in	
7	confide in	woo	confide in	confide in	confide in	
6	woo	woo	woo	woo	confide in	
6	woo	confide in	confide in	confide in	share interest	
6	confide in	confide in	confide in	share interest	confide in	
5	pick-up line	confide in	confide in	confide in	share interest	
5	woo	confide in	confide in	confide in	confide in	
16	woo	woo	woo	woo	woo	woo
10	confide in	confide in	confide in	confide in	confide in	confide in
5	confide in	confide in	confide in	share interest	share interest	confide in
4	confide in	confide in	woo	confide in	confide in	confide in
3	ask out	woo	woo	woo	woo	woo
3	woo	woo	woo	woo	woo	physical flirt
3	pick-up line	confide in	confide in	confide in	share interest	share interest
3	woo	confide in	confide in	pick-up line	confide in	confide in
3	confide in	woo	confide in	confide in	confide in	share interest
3	woo	confide in	confide in	confide in	share interest	confide in

Given the range of possible social exchanges the players could have chosen, the ones present in these sequences are overwhelmingly in line with the story goals of Simon's story, which are listed in chapter 4's Story Goals section. To revisit, the story goals are composed largely of involving Simon in friendship and dating relationships. Enemies and negative social exchanges are present in the story goals but to a lesser degree.

Out of the sequences shown, almost all contain social exchanges that have intents that, if accepted, would make progress to story goals. The only exceptions are the sequences that have only intents to increase romance network values (Woo and Physical Flirt). The rest (particularly those that include Confide In) are sequences that are aimed at completing story goals. Even the social exchanges in these sequences that do not directly address a predicate in a story goal condition change the social state to make a completed story goal more likely. Examples are the sequences with Back Me Up! or Share Interest exchanges followed by Confide In or Woos and Physical Flirts followed by Ask Out or Pickup Line.

Another strong indicator of strategic play found in these sequences are the lack of social exchanges that would reverse story goal progress or make story goals less likely to be completed. There are no social exchanges that would end friendships, end dating, or lower buddy network values like Txt Break Up, Bicker, or Blow Off Plans in these sequences.

Some of the most common sequences are made up entirely of Woo social exchanges. They are interesting because they do not indicate play that is aimed at completing story goals. However, they potentially indicate strategic play on part of the player. To create a sequence like this, it likely that the player abandoned pursuit of story goals in favor of their own personal goals; it may be that playing match-maker with *Prom Week* characters is fun and compelling gameplay. An alternative and less likely interpretation is that players were attempting the goals involving dating and did not understand the interface well enough to choose other social exchanges that would result in dating.

1.3.3 Social Exchange Frequency

To compliment the deep look at sequences in Simon's story, the rest of the section features a look at the frequency of social exchanges used by players in each story. By comparing social exchange frequency with a story's goals, a general estimate of strategic play across levels and players can be seen. Each story will be analyzed in the order they are unlocked by players. Figure 28 and Figure 29 contain a social exchange frequency graph for each character's story.

Table 19 – Doug's story goals.

Name	Condition
Big Buddy Doug	Doug makes a new friend in someone.
Super Nice Guy	Doug did at least 2 generally positive things.
Kinda Mean Dude	Doug did at least 2 generally negative things.
Last Minute Date	Doug was in a new dating relationship.

Doug. Doug's story is first and has the least complex and easiest to achieve story goals (his goals can be seen in Table 19). As shown in Figure 28, the social exchanges with the highest frequency (Woo, Annoy, Confide In, and Share Interest) fit nicely with the conditions of the story goals. This is to be expected as the story was designed to be easy,

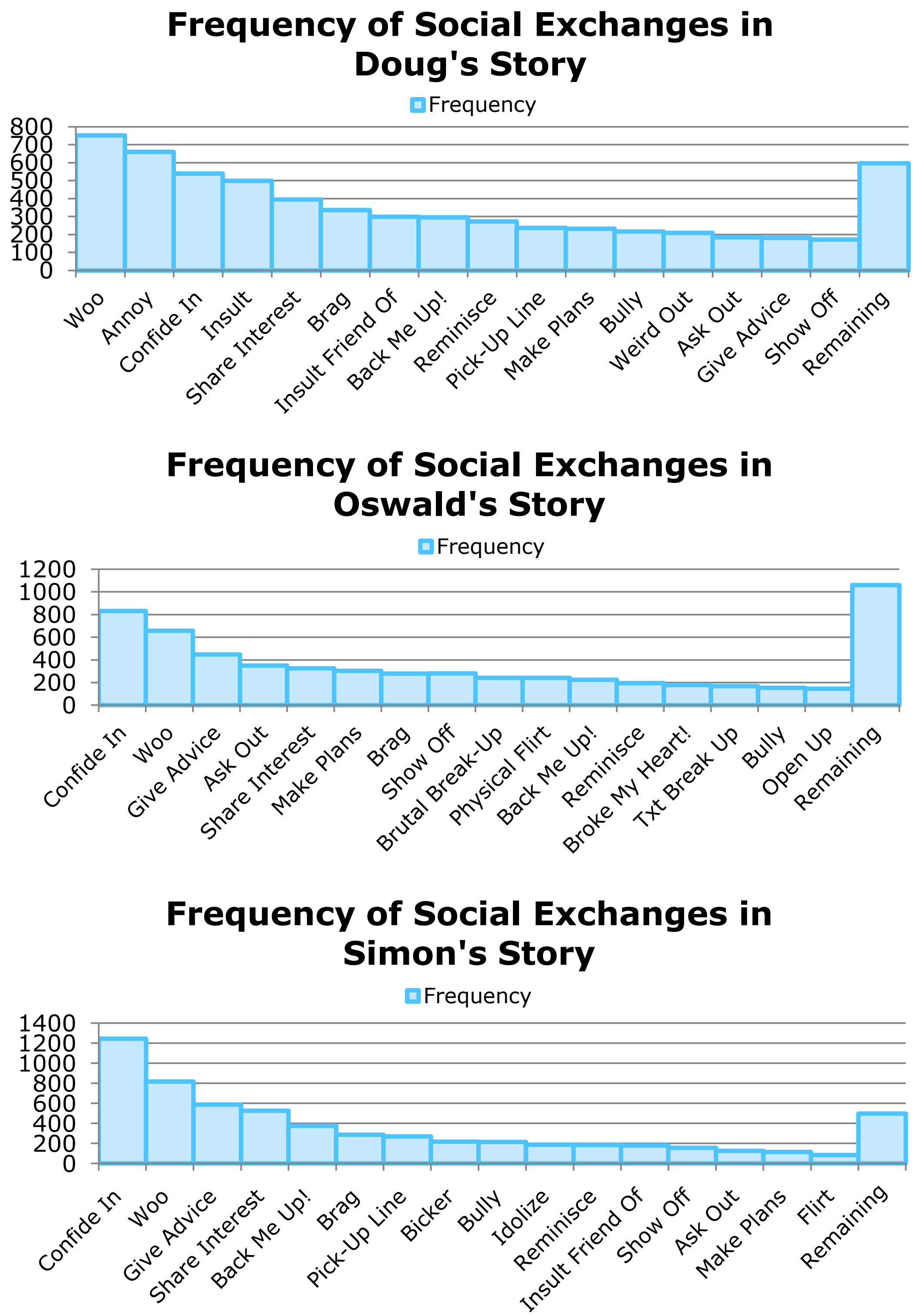


Figure 28 – The frequency of social exchanges for Doug's, Oswald's, and Simon's stories.

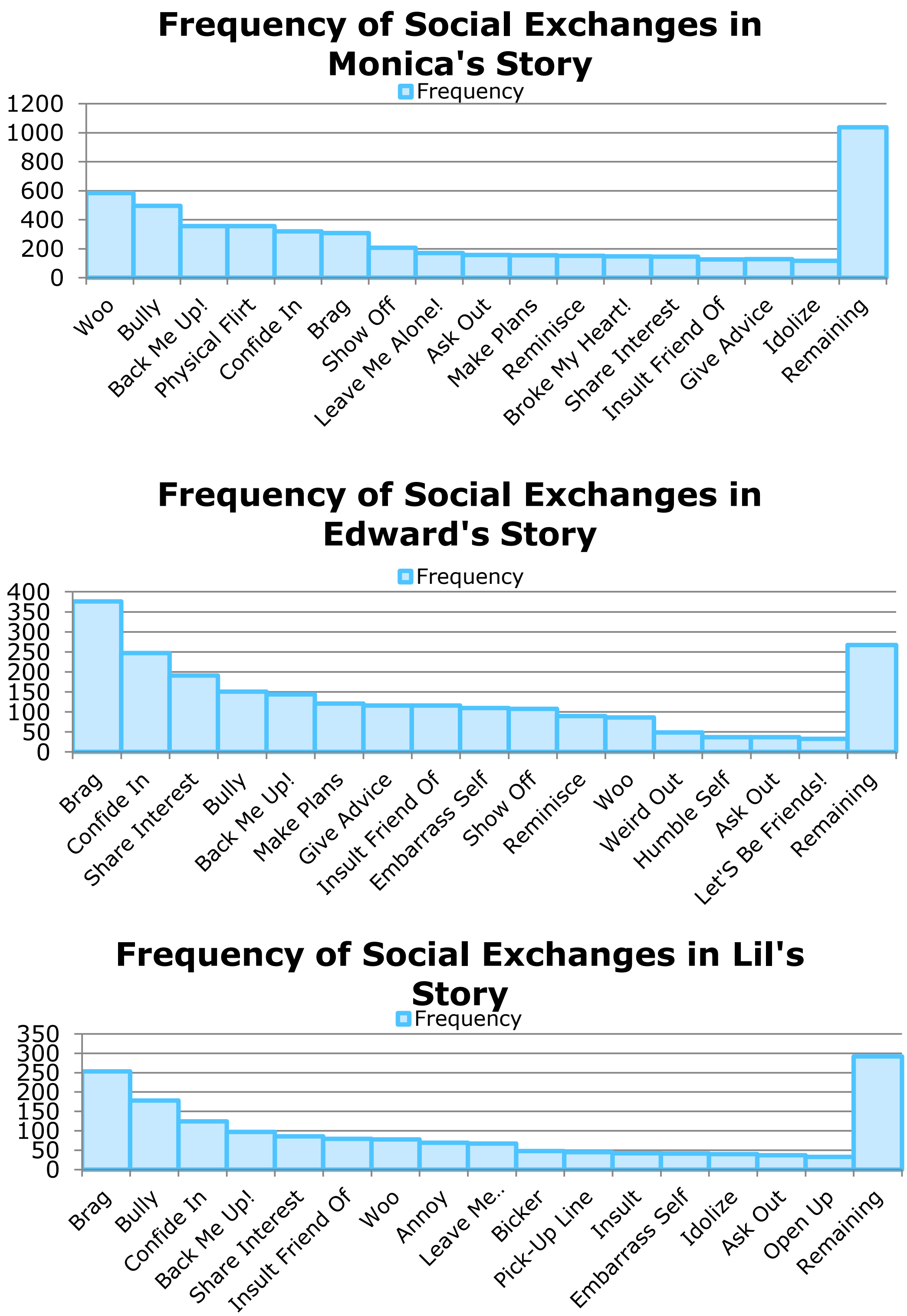


Figure 29 – The frequency of social exchanges for Monica's, Edward's, and Lil's stories.

has levels with small casts (which reduces the player's gameplay options), the casts are tailored to the goals (his first level has characters that will date or become friends with Doug in one social exchange), and the story goals have conditions with only one predicate each.

Table 20 – Oswald's story goals.

Name	Condition
Double Date	Oswald is dating someone. Someone is popular. Buzz is dating Naomi. They are the same "someone."
It's Complicated	Oswald is friends with someone. Oswald is dating someone. Oswald and someone are enemies. They are the same "someone."
The Better Man	Nicholas is friends with at least 4 people. Oswald is friends with at least 5 people. Nicholas is not dating Kate. Oswald is Dating Kate.

Oswald. Oswald's level is interesting in that it breaks the difficulty curve of the stories in *Prom Week*[30]. A partial cause of the extra difficulty comes from the jump in complexity of story goals (as seen in has levels with small casts (which reduces the player's gameplay options), the casts are tailored to the goals (his first level has characters that will date or become friends with Doug in one social exchange), and the story goals have conditions with only one predicate each.

Table 20) and the restriction of some story goals to a specific character ("Buzz is dating Naomi"). However, the frequency graph (Figure 28) shows that players worked toward the

[30] Bug fixes right before release destroyed the tuning on Oswald's story making it too difficult for the second non-tutorial story.

goals despite the difficulty. They chose social exchanges that had positive intended impacts on the friends relationship, buddy network, and the romance network which coincide with the conditions of all of Oswald's goals. The frequency graph also reflects that the player needs to make enemies to accomplish "it's complicated" and potentially break Nicholas and Kate up to accomplish "The better man" by having those games appear with a frequency greater than exchanges unrelated to the story goals but less than those associated with other story goals.

Table 21 – Simon's story goals.

Name	Condition
Totally Popular	Simon is friends with at least 5 people.
L0V3 in the Air	Simon is dating someone.
Caught Cheating	Simon is dating at least 2 people. Simon is cheating.
An Ideal Rival	Simon is friends with someone. Simon and someone are enemies. They are the same "someone."
Pitiful Me	At least 2 generally negative things happened to Simon.

Simon. Similar to the analysis over Doug's story, Simon's story goals are straight forward. Their conditions mostly consist of creating relationships. The frequency of social exchanges points to a simple correlation between the relationships required by story goals (friends in Totally Popular and dating in both L0V3 in the Air and Caught Cheating) and the social exchanges that cause those same relationships (Confide In, Pick-Up Line).

Table 22 – Monica's story goals.

Name	Condition
All About The Love	Monica is dating at least 3 people.
School Enemy #1	Monica has at least 6 enemies.
Life at the Bottom	Monica has no friends.

	At least 5 people think Monica is totally uncool.
Dark Angel	Monica is friends with Edward. Monica is friends with Mave. Monica is friends with Gunter.

Monica. The matching of story goal conditions with the frequency of social exchanges is discernible but not as clear as it was for the previous stories. This is due to the story goals requiring a broader range of social change across the board; dating, gaining friends, losing friends, gaining enemies, and cool network down are all called for. The players still played strategically toward the story goals as evidenced by the top ten most frequent social exchanges only having two outliers: Brag and Show Off, both of which increase cool network values. The rest of the top exchanges have intents that would make goal progress.

Table 23 – Edward's story goals.

Name	Condition
Burning Bridges	Edward is single. Edward has at least 5 enemies.
The Bright Side	Edward is friends with at least 5 people. Edward is dating someone. Edward did something nice.
Dark is the New Black	At least 5 people think Edward is cool. Mave is friends with at least 5 people. Buzz is friends with fewer than 2 people. Monica is friends with fewer than 2 people.

Edward. The correspondence between story goal conditions and frequency of social exchanges aimed at those conditions is clear in Edward's story. The players chose social exchanges that would result in friendship (Confide In and Make Plans) and increased buddy network values (Back Me Up!, Share Interest, Give Advice, and Reminisce) which coincides with The Bright Side and Dark is the New Black. Dark is the New Black also corresponds to the highest frequency social exchange Brag, which has a cool network up intent. Bully and

Insult Friend Of, both social exchanges that begin enemy relationships, were often chosen and match Burning Bridges. The only outlier is the cool network down social exchange, Embarrass Self. Interestingly, Edward's story is the strongest of the stories in displaying evidence for strategic play; not even the early levels are as strong.

Table 24 – Lil's story goals.

Name	Condition
Queen of Chic	Lil is dating someone. At least 4 people think Lil is cool.
Out with a Bang	Lil has at least 6 enemies.
Clean Break	Lil has no friends. Lil has no enemies. Lil is single.
Emo Angel	Lucas is dating Phoebe.
Sabotage the Populars	Buzz and Nicholas are enemies. Buzz is not dating Cassandra. Monica and Cassandra are enemies.

Lil. In contrast to Edward's story, Lil's story shows little correlation. With story goals that, in total, require nearly every relationship to both begin and end, Lil's story has a structure that thwarts attempts at social exchange frequency analysis. Additionally, Lil's story is the last story to be unlocked in *Prom Week* and has many less play traces than any other story. This story is inconclusive in finding evidence of strategic play in *Prom Week*'s player traces.

Given both the analysis of sequences in Simon's story and the analysis of social exchange frequency of all the stories, the player traces clearly contain evidence of strategic play. With either the guidance of story or self-imposed goals, players are able to manipulate the social state in directed ways via social exchange play.

2 Authoring Complex Spaces

This section provides evidence for this dissertation's second research contribution, making authoring possible for a new level of player choice complexity. The complexities of *Prom Week* will be compared with the current state of the art in story-centric CRPGs. The CRPGs in this comparison are the *BioWare* games *Mass Effect 3* (BioWare 2012)and *Star Wars: The Old Republic* (*SWTOR*) (BioWare 2011). *BioWare*, a game development studio that specializes in CRPGs, is often touted as releasing state-of-the-art story-driven games[31]. Their *Mass Effect* series combines action-oriented combat with story interaction via dialogue trees. *SWTOR* is a modern massively multiplayer online role playing game (MMORPG) that is unique in bringing the story-centric style of the *Mass Effect* series into the space of MMORPGs.

As both *Mass Effect 3* and *SWTOR* allow for player choice in story via dialogue trees, the lines of dialogue provide a starting point for comparing complexities. The three games in the *Mass Effect* series have many lines of dialogue: the original *Mass Effect* (BioWare 2007) has

---

[31]The following video game reviews extol the virtues of *BioWare*'s story-centric CRPGs:
http://www.gamingnexus.com/Article/Mass-Effect-3/Item3417.aspx
http://www.pcgamer.com/review/dragon-age-2-review/
http://www.g4tv.com/games/pc/61502/star-wars-the-old-republic/review/

20,000, *Mass Effect 2* (Bioware 2010) has 25,000, and *Mass Effect 3* has 40,000 lines of dialogue[32]. *SWTOR* breaks dialogue content records by containing over 200,000 lines of dialogue[33].

These massive numbers do not translate automatically into large amounts of player choice. Portions of these numbers are claimed by combat, idle NPC, and utility (e.g. speech from vendors) dialogue. The remaining portion is further winnowed by segments of non-interactive dialogue (consecutive lines with no options for player choice) and interactive dialogue where the player's choice has little or no bearing on either successive dialogue, the state of the world, or future game events. What is left is in dialogue trees with meaningful choices, which will be referred to as choice dialogue.

Another mitigating factor restricting player choice with respect to lines of dialogue in all CRPGs is the amount of side quests that largely have no effect on the story told by the game. Based on walkthrough documents[34], *Mass Effect 3* has 15 story quests, 6 multiplayer

---

[32] These estimates of lines of dialogue come from information released by *BioWare* via social media and by scripts compiled by fans of the game. Video games media has covered these numbers:
http://beefjack.com/news/mass-effect-3-has-40000-lines-of-dialogue/
http://www.gamersbook.com/scene/news/mass-effect-3-features-40000-lines-of-dialogue/

[33] http://www.pcgamer.com/2012/01/06/star-wars-the-old-republic-scoops-guinness-world-record-for-voice-acting/
http://www.joystiq.com/2012/01/07/star-wars-the-old-republic-wins-guinness-award-for-most-voice-a/

[34] The numbers of different quest types are estimated from the following walkthroughs:
http://www.gamefaqs.com/xbox360/995452-mass-effect-3/faqs/63886
http://www.ign.com/wikis/mass-effect-3
http://masseffect.wikia.com/wiki/Mass_Effect_3_Guide

quests, and 39 side quests (a few of which affect the future story). This ratio of quest types is indicative of the beads-on-a-string model mentioned in chapter 1. This model is often used by CRPGs to restrict the story space explorable by the player into a manageable number of possible states.

While there are no available estimates of the amount of choice dialogue in *Mass Effect 3*'s 40,000 or SWTOR's 200,000 lines of dialogue, that number may be greatly diminished given the dialogue needed to support the complex combat system and the highly-detailed environment and finely realized soundscape of *BioWare's* games.

In contrast to the diminishing of the lines of dialogue into choice dialogue, *Prom Week*'s dialogue templates increase the amount of choice dialogue. *Prom Week* has 4987 NLG dialogue templates (described in chapter 3's Performance Realization section) that can be used by any combinations of characters; they are abstracted away from specific characters into performances usable by all characters. Unlike lines in dialogue trees that cannot be reused, the dialogue templates in *Prom Week* can be used whenever there is appropriate character design and social context.

This comparison emphasizes one of the strongest features of *CiF*; it is a system for dynamically retargeting dialogue. As was previously explained, *SWTOR's* total dialogue is reduced to a small amount choice dialogue. Each line of its choice dialogue is tied to specific characters and cannot be reused. In contrast, each of *Prom Week's* dialogue templates can be reused and retargeted for any set of characters where the appropriate social context is present. By abstracting social exchanges and their performances away from being acted at a specific time by specific characters, the a dialogue template authored for *Prom Week* is effectively multiplied by the large number of potential situations in which it can be used.

Considering the lack of player choice ascribed by dialogue trees in even the best of story-centric CRPGs, the player choice complexity enabled by *CiF* and present in *Prom Week* is at a level that was previously very difficult or intractable to reach with traditional methods. The amount of resources available to *Prom Week* for authoring content was relatively small when compared to most CRPGs; this new level of complexity will become even more apparent with more ambitious future projects that have access to more resources.

Chapter 6

Holodeck

CiF's social exchanges represent social interactions at the level of small conversation in small groups. The system also simulates the social world in discrete turns. While these qualities of *CiF* were useful as design constraints, their use left the corresponding areas of design space unexplored. From the perspective of the work in this dissertation, the *Holodeck* project addresses the areas not addressed by *CiF.* Specifically, real-time social interaction, non-verbal performance, social proximics, and rituals (such as greeting, being in a conversation, and group membership) are included in the research done in the *Holodeck* project. This research helped us to understand what shorter-term, non-verbal social exchanges where and how they could be represented in a system. The understanding of these types of social exchanges allowed for the creation of AI-controlled avatars that could interact with other avatar (including bots, other AI avatars, and players) in the *Holodeck* project. External to the main contribution of this dissertation, the *Holodeck* project is included this research in service of creating avatars that achieve a high level of supporting character realism.

1 Supporting Character Realism

In media such as video games, characters are often defined as *believable*, a term borrowed from animation and the character arts. Here, a successful NPC is not one that fools the player into thinking they are controlled by a human, but rather one in which the player willingly suspends disbelief (Coleridge 2006). Thus a believable NPC creates a consistent and compelling illusion of life (Loyall 1997), one that a human participant willingly accepts as a representation, without ever thinking about (or caring) whether the character is controlled by a human being. While believability can provide rubrics for the creation of NPCs, some training applications will require trainees to treat NPCs as if they were human, requiring NPCs that don't immediately highlight their representational nature.

In contrast, Turing Test realism assumes the adversarial conditions of the Turing Test (Turing 1950), in which a human participant actively probes to expose a computer masquerading as a human being. In this strongest sense of "realism", an NPC succeeds if it actively fools a participant into thinking it is human in the face of skeptical and probing interactions. This would likely require a complete artificial intelligence model of a human being, making this sense of realism useless for building compelling and effective NPCs now or even potentially in the long term.

We define supporting character realism (SCR) as a new category for determining NPC success. A supporting character is one which engages in background activity and light interaction with the player. A supporting character isn't intended to engage in long-term complex behavior with the player, but rather to provide a sense of realistic human activity around the player. Distinct from believable characters, which the player knows are not real but willingly suspends disbelief, and Turing Test realism, in which the player actively probes to determine if the character is computer or human controlled, with SCR the player doesn't think to question whether a character is human or computer controlled. In a mixed environment of human avatars and SCR agents, in the context of background activity and lightweight interaction with human-controlled avatars, an SCR agent blends in seamlessly. In the context of virtual world training environments, SCR is useful where many characters are needed, such as an urban street scene. If it is possible to build AI characters that achieve SCR, then human trainers would only need to control characters with which the trainee has complex and detailed interactions, while SCR agents fill out the rest of the space. Trainers may need to dynamically switch control as a trainee begins detailed interactions with a formerly background character. But with SCR a trainer would be able to control a large number of characters in the training environment.

To determine whether it is possible to achieve SCR, we developed AI-controlled prototype avatars and tested them in the context of a social scenario implemented in the

Second Life virtual world (Lindon Lab Inc. 2003). The social scenario, set in a bar, is inhabited by a mixture of human controlled avatars, AI-controlled avatars (SCR avatars), and bots, where the bots are implemented using standard *Second Life* scripting techniques. We first established baseline metrics by which to compare how participants relate to avatars in the virtual world. Then, with human, bot and SCR avatars all interacting and role-playing bar denizens, participants were asked to engage in simple social interactions tasks. If we are successful in implementing SCR, the SCR avatars should lie between bots and human-controlled avatars along the various interaction metrics. And indeed, analysis on the gathered data indicates that SCR avatars are much closer to humans than to bots in the majority of SCR metrics.

We build on prior research in our definitions of SCR behavior, quantitative and qualitative metrics for characterizing avatar interactions, and in our technical infrastructure and approach for integrating external AI-controlled characters into *Second Life*. We used *Second Life* as the virtual world environment for exploring SCR. In addition to boasting a substantial player population, *Second Life* offers a virtual environment with research-friendly capabilities that include computer controlled avatars, data recording and very customizable environments. Varied research topics including education (Bell 2009), simulating fire accidents (Buono et al. 2008), and modes of human-computer interaction (de Pascale, Mulatto, and Prattichizzo 2008) have been conducted in Second Life.

Second Life and other virtual worlds have been used as a methodological tool to study human interactions with both other human and AI controlled avatars (Blascovich et al. 2002). Studies in the social behavior of players, such as accrual of social capital (Huvila et al. 2010), gender and sex practices in virtual worlds (Brookey and Cannon 2009), social affordances of players (Zebrowitz 2009), decoupling of non-human behavior between players (Buono et al. 2008), and effects of gender differences between a human and their avatar (Nick Yee et al. 2011) provide particularly useful tools for both defining the behavior of SCR avatars and developing metrics for characterizing avatar interactions.

We borrow some of our spatial measures from Friedman et al (Friedman, Steed, and Slater 2007). This work makes use of bots written in the *Second Life* scripting language that explore and find objects of interest. The bots recorded information about player's special responses to other avatars and their proximity in dyadic interactions. Given the additional capabilities of our SCR avatars vs. the rather simple behavior supported by the *Second Life* scripting language, our study supplements these spatial metrics with metrics such as gesture frequency, avatar facing, and a questionnaire about co-presence.

How social behavior changes over time and expressions of personality in virtual worlds were analyzed in a longitudinal study of a group of around 80 students enrolled in a class about *Second Life* (Harris et al. 2009; N Yee et al. 2010). A framework to gather avatar-related information from *Second Life* over extended periods of time (Nick Yee and Bailenson

2008) was used in conjunction with qualitative measures to perform both analyses. Social involvement was shown to increase over time through similar increases in metrics of number of friends, groups, and time spent in populated areas. The subjects' activity and exploration was shown to decrease over time with increases to low-energy actions and time spent in their 3 favorite regions while the metrics of unique regions visited, teleports used and high-energy actions decreased. Qualitative data in the form of weekly questionnaires demonstrated that the distribution of types of activities the students engaged in stabilized over time and was consistent with how time is spent in the real world. Expressions of personality were studied using the measures of a 50-item scale measuring the Big Five factor structure, and avatar-related metrics, which include stance, frequency of logging in, and nearby avatars, and linguistic measures. These behavioral metrics were shown to have high rank-order and low absolute stabilities while very little stability was seen in the linguistic measures. Correlations of conscientiousness and emotional stability with the behavioral and linguistic measures were found. Although our methodology to evaluate SCR is similar, the salient portions of the collected data are different as SCR is evaluated in a single environment with a focus on dyadic interactions between the participant and other avatars. Additionally, since we include human, bot and SCR avatars, different methods of data analysis are necessary to determine the trends of participant's interactions with respect to avatar control type.

Weitnauer (Weitnauer et al. 2008) makes use of a similar AI architecture to the one we employ to create a proof-of-concept implementation of an AI avatar with more capability than standard *Second Life* bots. We make use of similar technical infrastructure in using the the *libOpenMetaverse*[35] framework to connect our AI system and data reporting framework to *Second Life*. Weitnauer's work was a technical proof of concept, and thus was never evaluated with respect to human interaction.

2 Methods

In order to experimentally operationalize SCR, we constructed a social environment that encourages constant interaction and is filled with distinguishable social roles and personalities that would be recognizable to our participants, with supporting characters being played by a mixture of humans, bots and SCR avatars. With these goals in mind, we chose a virtual bar for our environment, defining five roles in this setting. Our first experiment, an observational study with humans and simple computer-controlled scripted bots, was constructed to establish the metrics for measuring and differentiating between the actions and interactions of human and computer controlled avatars, and to provide behavioral targets for the SCR avatars based on how our confederates played their roles

[35] http://lib.openmetaverse.org

2.1 Scenario and Roles

In our experiments, a single participant is introduced to our virtual bar filled with both human-controlled avatars (henceforth referred to as “confederates”), and computer-controlled avatars, which are either traditional scripted bots or prototype SCR avatars, both described in following sections. The participant takes the role of a student entering the bar for the first time. In the observational study human confederates and bots performed the following social roles commonly found in bars:

Regular: The veteran of the nightclub/bar. The regulars have their frequented hangout locations and know all of the rules of form of the bar. If someone breaks those rules, the regulars are the first to correct the violators.

Spring Breaker: This gregarious personality is always doing something; dancing, running to the bar for more drinks, bringing others into dance groups, running to up to strangers and getting a groove on: the life of the party.

Wallflower: In social situations, a wallflower is a slang term used to describe shy or unpopular individuals who do not socialize or participate in activities at social events. It is most often used to describe someone who stays close to a wall and out of the main area of social activity.

Waiter/Waitress: As the service staff of the bar, the waiter or waitress caters to the patrons of the bar by serving drinks, and attends to the bar and the various tables around the dance floor.

Generic: Our generic personality embodies a passive but friendly patron that blends into the general activity of the bar. He or she hangs around in various places around the dance floor with a drink in hand, observing the activity around and reacting in kind to dancing and positive gestures from other patrons.

Participants were tasked with various social interactions with patrons of our virtual bar, with the intention of them interacting with every avatar in the virtual bar. These tasks included: getting to know other avatars at the bar, finding the shyest/most outgoing/most popular patron in the bar, buying somebody a drink, hanging out with various social groups, and getting an avatar to buy the participant a drink.

2.2 Non-linguistic Interaction

In our experiments, we chose to exclude the voice and chat capabilities of the *Second Life* platform, and limit interaction between all avatars in our scenario to physical movement, gestures, dancing, sitting/standing, and giving/receiving drink objects. As previously stated, the goal of this project is to explore the physical and embodied aspects of social interaction, which would typically not include the socially demanding, subtle, and interrogative interactions of chat. While future work will focus on limited supporting character language capabilities, we focused here on physical interaction and performance. In order to establish a credible reason for this limited interaction, our virtual bar environment was constantly bombarded with loud dance music, like that of many real-world bars or clubs.

Figure 30 - Our test setting consisted of a furnished virtual bar, complete with loud music.

3 Second Life

We used *Second Life* as our virtual world platform, due to the platform's existing technical support for networked multi-user interaction, expressive avatar gestures, scripted data logging and computer control of avatars. Additionally, *Second Life*'s large virtual world has an extensive market of virtual assets, including additional avatar gestures, avatar clothing, buildings, furniture and props that allowed us to easily shape our virtual bar into a more plausible environment filled with visually distinguishable avatars.

Our virtual bar itself was purchased from a vendor in *Second Life* and populated with objects commonly found in bar settings. Posters, dart boards, sofas, tables, lights, a dance stage, and stock of drinks are all present in the experiment's virtual environment. As seen in Figure 30, the bar consists of areas appropriate for the waiter/waitress (behind the bar), places for the regulars and their associates to group (tables, bar stools, sofa), out of the way locations for the wall flowers (places to sit in the corners and pools to lean on), and an area for showing off and dancing (the dance floor).

4 Observational Study

We began with an observational study, a non-intrusive evaluation which placed seven participants with confederates and traditional scripted bots in our scenario. These participants were given the task list described above, and given unlimited time to complete these tasks. In order to account for gender biases affecting the perceived personality and interactions of the various patrons of the bar, the gender of each avatar, as evident by their physical appearance and clothing, was set based on the gender of the participant, either 'same' gender as the subject, or 'opposite' gender from the participant. This study included eight bar patrons, with three human-controlled regular avatars, one human-controlled spring breaker avatar, a human and computer controlled waiter/waitress, and a human and computer controlled wallflower.

Each of our confederates were asked to freely role-play an assigned social role in our virtual bar using all available interactions except chat, while our computer-controlled avatars used a fixed set of sequential actions based on these roles. The protocol followed by this study consisted of the following procedure: (1) Participant reads and signs informed consent document, (2) *Second Life* tutorial and overview of the participant task list, (3) Setup virtual world with confederates and computer-controlled avatars, and enable data logging, (4) Perform experiment in *Second Life*, (5) Debriefing interview, and (6) Survey.

Using video recording and automatic data logging via in-game logging scripts, we observed and analyzed the interactions amongst all avatars, both human and computer-controlled. After running our experiments, we viewed the captured video and noted actions or style of actions that were significantly different toward bots than toward human avatars, and analyzed our data logs using various metrics gathered from previous virtual world interaction research. These metrics, discussed in detail in our results section, include interpersonal distances and space categories, gaze angles and sums, and measures of presence and co-presence. This initial study allowed us to confirm that there were measurable, significant differences between human-human and human-computer avatar interaction in our scenario, motivating our further experiments. In Figure 31, we show a spectrum of Avatar Realism that we hypothesized would exist, ranging from the realism of a traditional scripted bot to that of fully human-controlled avatar. In our research, we expected that our metrics would place an agent with effective Supporting Character Realism in between these two ends of the spectrum.

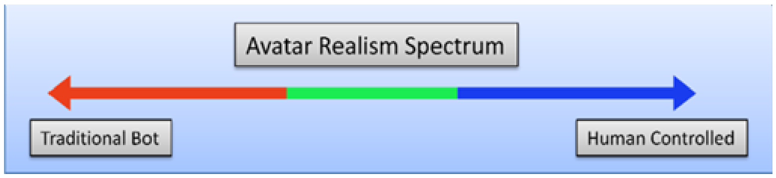


Figure 31 - The spectrum of Avatar Realism. We expect the metrics of an agent with effective SCR to be placed between the realism of a traditional bot and that of a human controlled avatar.

Our observational study, in addition to establishing a concrete way to measure the differences in interaction between computer and human controlled avatars, provided a large catalog of recorded behavior of humans enacting social roles in our scenario. We used this information to design computer-controlled avatars that engage in the same behavior. In our study, we placed these AI-driven avatars alongside confederates and bots in the same social scenario as our observational study, in order to evaluate their realism and place them somewhere in the spectrum shown in Figure 31.

5 Architecture

To develop a prototype SCR avatar, we leveraged existing frameworks and APIs to make rapid research progress. We built on the *A Behavior Language* (*ABL*) autonomous agent architecture (Mateas 2002; McCoy and Mateas 2008; Mateas and Stern 2002; Mateas and Stern 2004) for behavior control and took concepts from *CiF* for social exchange management. In the following subsections we provide details on the technical design and infrastructure for our prototype, and on the behavior building for three distinct personality types: the regular, the spring breaker, and the waiter/waitress.

The Holodeck project implementation is based on a variety of languages, with an agent architecture that combines behaviors written in the reactive planning language *ABL* that compiles to *Java*[36], communicating with a Second Life interface written in *C#*[37], along with a study-time data logger written in *LindenScript*[38] communicating to a central data storage server written in *Node.js*[39].

5.1 Study-time Data Logger

For each participant that ran through our observational, pilot, and full studies, we would record various metrics for later analysis. Logged once every second, these metrics included physical position, velocity and rotation of all avatars in the scenario space, as well as currently running animations and collision information from the last one-second window of time. This data reporting was accomplished using a custom data gathering script written in the *LindenScript* language, and was attached to every Second Life account

[36] http://java.sun.com

[37] http://msdn.microsoft.com/en-us/vstudio/default

[38] http://wiki.secondlife.com/wiki/LSL_Portal

[39] http://nodejs.org/

involved in each run of our study. These scripts awaited specific textual commands received over a public chat channel by known administrator accounts. Each time the study would begin or end, an administrator would broadcast a command to activate or deactivate all data logging scripts. During the study, every script would detect all required information and send it to a remote logging server through an HTTP GET[40] request with a series of query parameters. This remote server, written in the *Node.js* language, took each request, parsed the query string and appended the data logging contents of each HTTP request as a comma-separated row in a flat text file, for later conversion and analysis. Our logging script was limited to a one-second reporting rate by the *LindenScript* internet-access-throttling facilities imposed by the Second Life servers.

[40] http://www.w3.org/Protocols/rfc2616/rfc2616.html

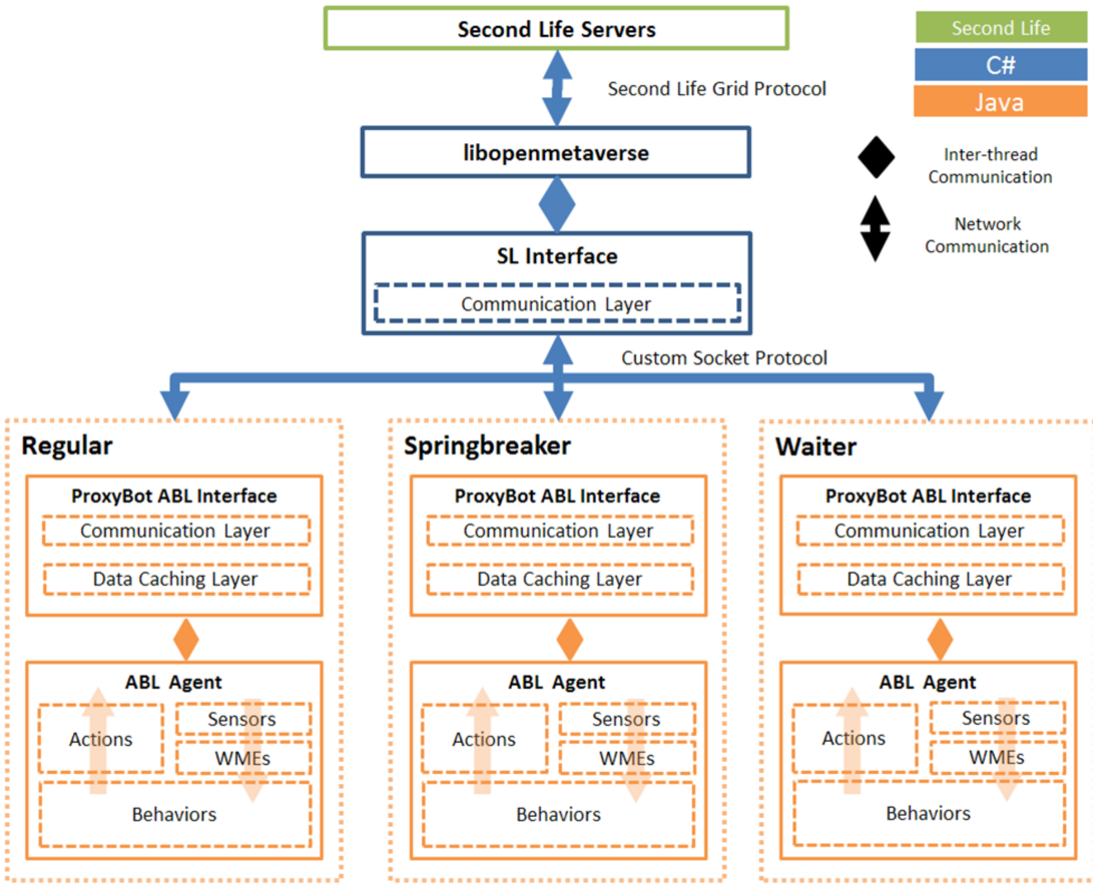


Figure 32 - The SCR avatar architecture for the Holodeck project.

5.2 Bots

For our studies, we wished to compare the performance of human confederates and SCR avatars with a traditional scripted bot found in the background environment of many modern game engines and virtual worlds. The publicly available bots in the *Second Life* market are typically primitive vending machines for drinks or dances that wait for *Second Life* users to choose from a scripted menu of items, and may print out a static piece of text for new visitors, or repeated pieces of text for existing guests. Because these primitive existing bots did not have any concept of sustained and active interaction with people around them, we implemented our own bots to represent a typical scripted bot from other game environments. For the observational study, we implemented two bot roles, the wallflower and the waiter/waitress, and then added a "Random" role to our pilot and full studies, which was meant to represent a typical bar-goer that blends into the background and occasionally interacts with those around him or her.

The *Holodeck* bots, unlike the *ABL*-based SCR avatars, were functionally simple and did not require the language and infrastructure features of *ABL*. As a result, the bots were implemented as a scripted loop of sequential actions in a single method, stored in a separate class in the *C#* code base for the *Holodeck Second Life* interface program. These bot behaviors were then activated and connected to a *Second Life* account when the study began. They were programmed to continuously loop over the set of actions, using a simple condition and a random number to select if a specific action would trigger. For example, the wallflower would check to see if any person was close to him or her, and immediately turn and use one of two randomly selected negative gestures to ward them off. The waiter/waitress would look for potential customers close to him or her and play a random friendly gesture and wait, while the Random bot would turn toward people close to him or her and randomly choose to dance, laugh, clap, or nod. In addition to these random actions, each bot role was given a manually-created set of potential fixed places in the world it could be, and would walk between those locations as part of its action selection process. For the wallflower, the set of locations were places to sit down, as well as target locations to move to if anybody got close at the current location. The waiter/waitress bot had a list of places to stand and wait for customers to approach, and the Random bot had a list of various spots in the bar that people would typically hang out, where he or she would fit in.

5.3 SCR avatars

The SCR avatars were authored in *ABL.* They are composed of behaviors that describe computation and actions performed by the agents and working memory elements (WMEs) that store both internal state for the agent and input from the environment in which the agent operates. Behaviors contain a precondition, which must evaluate to true before a behavior can be executed, and a series of steps, which can be primitive actions that effect the environment, mental acts that perform internal calculation and WME management (editing, deletion, or addition), and subgoal steps. Subgoal steps call one or more behaviors with the same name and parameter list, evaluating all preconditions and selecting amongst the set of all behaviors for that goal that have valid preconditions. Behaviors can be executed sequentially, in which every step of the behavior is performed in order, or in parallel. All *ABL* agents begin with an initial tree of parallel goals that activate behaviors that can in turn subgoal to other behaviors. In this way, an agent and its corresponding hierarchical arrangement of tasks is said to have a behavior tree and various behavior subtrees within the tree.

Our agent architecture used for the SCR avatars takes as input various state and events from *Second Life* and turns them into WMEs in *ABL*. These inputs include: state updates for objects, avatars and self which contain name, social role, position, velocity, and rotation information. This state information is updated regularly multiple times a second, and stays in memory for as long as the agent is run and the object or avatar exists in the world. Additionally, when an animation is activated by any avatar, or an object is given to the avatar, events are generated and stored as WMEs. These event-based inputs stay in memory for a fixed amount of time (on the order of 5-10 seconds) before being wiped from memory. If no behavior identifies and disposes of these events within that time frame, it is assumed that the events were not important to the behaviors in our avatar.

SCR avatars can perform a series of primitive actions from within their *ABL* behaviors. These actions include: walking to a specific location, named avatar, or named object; turning toward a specific location, named avatar, or named object; playing a named gesture stored in the gesture palette; giving an item to a named avatar; wearing an item stored or given to the avatar by another; removing an item from the outfit of the avatar; standing; sitting; touching an object (pointing at and activating the ‘beam’ animation effect); and looking at an object (locking the head direction at a specific named object in the environment).

Through analyzing the videos from the observational study via dramaturgical analysis, we unearthed the social exchanges that are consistent with the real-time, non-verbal constraints of the project while representing patterns of social interaction. The *ABL* behavior trees that drive SCR avatars are divided into branches that roughly encapsulate social exchanges. Some examples of social exchanges for the SCR avatars are the DanceCircle and ChatCircle behaviors. They are abstract patterns of behavior that are reified by sub-behaviors into concrete performances (consisting of socially appropriate gestures, character facing, timing, and proximity) consistent with the current social state in real time.

In the diagram below, we show a sample of truncated and simplified *ABL* code that drives a specific behavior in our SCR avatars. In this sample, we describe a behavior called ChatCircleBehaviorTree() that describes a parallel behavior for interacting with other Second Life avatars in a conversation/chat circle. This behavior simultaneously executes three subgoals that perform the following tasks: 1) ensure the avatar occasionally adjusts his or her physical position and rotation like a human-driven avatar would (HumanLocationAdjustment()), 2) ensure that the avatar maintains an appropriate distance from other avatars (MaintainMinSocialDistance()), and 3) monitor the gestures and positions of the other avatars in the chat circle and respond to them (ChatCircleGestureMonitor()). This parallel behavior also contains a context condition, which is a precondition that must evaluate to true as long as any steps within the behavior are executing. This constantly-true precondition differs from a standard precondition, which must only evaluate to true during the behavior selection process. If the context condition ever evaluates to false, then all steps (including subgoals) of this behavior are stopped from executing. In this behavior, the context condition simply ensures that the WME of type ChatCircleWME exists in the working memory store. We use this type of context condition to quickly switch between sets of behaviors by simply removing these specific types of WMEs from working memory in other behaviors which are built to manage these transitions.

In the sample, the first variation of the behavior ChatCircleGestureMonitor() contains a precondition that looks for a friendly gesture event from another avatar that is within a specific distance threshold called PERSONAL. This requires the precondition to first grab the state information for both the avatar executing this behavior (stored in the SelfWME object), as well as the state of another avatar. This information is then compared in the third precondition statement to ensure the maximum distance requirement is met, and the final step of the precondition locates an AnimationWME object that stores the information for an animation event that came from the avatar within that distance threshold that is also considered a friendly gesture and directed at this avatar (with a gaze angle less than 45 degrees). If any one of these precondition statements does not validate, then we know that there are no avatars within the specified distance that are also gesturing in a friendly way toward our avatar. Finally, this behavior takes the information captured in the precondition (using the :: binding operator), and passes it to a subgoal that dictates how to respond to the incoming friendly gesture. In our behaviors, we used the concept of gaze angle, gaze sum, and interpersonal distance thresholds (Nick Yee et al. 2007).

The following is a sample of *ABL* code:

```
parallel behavior ChatCircleBehaviorTree() {
    context_condition { (ChatCircleWME) }

    subgoal HumanLocationAdjustment();
    subgoal MaintainMinSocialDistance();
    subgoal ChatCircleGestureMonitor();
}

sequential behavior ChatCircleGestureMonitor() {
    precondition {
        self = (SelfWME)
        target = (AvatarWME name::tName)
        (AvatarWME.withinSocialDist(SocialDist.PERSONAL, self, target))
        anim = (AnimationWME avatarName==tName gazeAngle < 45.0
        animationName::animName isFriendlyGesture==true)
    }
    ...
    subgoal ChatCircle_RespondToFriendlyGesture (tName, animName);
}

sequential behavior ChatCircleGestureMonitor() {
    ...
}
```

At the top, ChatCircleBehaviorTree() is a parallel behavior that subgoals three behaviors which execute simultaneously. When subgoaling to

ChatCircleGestureMonitor(), *ABL* checks the preconditions of all behaviors named ChatCircleGestureMonitor(), searching for satisfied precondition bindings and constraints before choosing a behavior to execute. If a precondition is not satisfied, that behavior is removed from the list of potentially selected behaviors. Above, the first variation of ChatCircleGestureMonitor() detects an avatar that is gesturing in a friendly way towards us (gazeAngle < 45.0) and is within the PERSONAL interpersonal distance threshold.

For our agents, the *ABL* language has provided many powerful features that make it ideal for use in our SCR avatars. These features include powerful precondition search and binding, parallel behaviors, and continuously monitored conditions. In the above sample code, the precondition in the second behavior uses a series of constraints and chained bindings that would require a verbose and quite large quadrupally-nested loop in any traditional procedural language such as *C*, *C++, Java*, or *C#.* Additionally, with a single keyword parallel, we were able to specify that a behavior be executed in parallel, without requiring manual management of execution ordering, memory sharing, and resource contention. Finally, *ABL* is built around being a language that can monitor and quickly react to input from the environment, and as such, has simple syntactical mechanisms for specifying that an avatar's behavior wait for a specific condition to be true. The context condition in the sample code above constantly monitors the condition within it (that the

ChatCircleWME object must exist in memory), without requiring any manual multi-threading code or safe memory access management. Similarly, *ABL* provides a **success_test** construct that acts like a context condition, but can be placed anywhere in the steps of a behavior, pausing execution of a behavior until a precondition (with associated bindings) is met. All of these features represent a unified competency for real-time reactive behaviors that execute together and interact fluidly with each other. Combined with a familiar *Java*-like syntax, *ABL* proved to be an extremely effective language for the SCR avatars, allowing for quick development, powerful yet succinct behaviors, and with enough built-in infrastructure to allow us to primarily concentrate on building intelligent and realistic avatar behaviors without the distractions of additional infrastructure requirements.

When designing the SCR avatars, we first observed the behaviors of confederates in the captured videos of our observational study. We made note of the major types of activities performed by all confederate roles and the specific types of actions and reactions that occur between these roles. As mentioned in the previous section, we used social exchanges as a framework for our analysis of this observational study data. Using these high level notes, we then structured our *ABL* agents to match our findings.

5.3.1 Common Behaviors

All of our SCR avatars share a set of common behaviors that we observed amongst all roles in our observational study. These behaviors are as follows:

MaintainMinSocialDistance. This behavior maintains a minimum interpersonal distance between the avatar and all others. If the avatar or another avatar passes the minimum threshold, then our avatar will step backwards away from the closest avatar near it.

HumanLocationAdjustment. This behavior emulates the occasional minor and noisy physical adjustment that confederates in our observational study performed throughout the study. Every five seconds, this behavior has the probability of adjusting the avatar in a random direction by stepping or rotating slightly. This helped our avatars to appear as if a human was driving them.

PlayGesture. Our primitive action to play a gesture simply activates the gesture immediately. We created this common PlayGesture behavior that would mimic the actions a human takes to activate a behavior in the *Second Life* viewer interface. Because *Second Life* uses the movement of the mouse to adjust the head rotation of an avatar, the playing of a gesture always rotated the head downward or to one side as the human controlling the avatar moved his or her mouse to the gesture menu. This behavior appears unusual and not realistic, but nonetheless is a common action that occurs when gestures are used. In order to mimic this behavior, our SCR avatars first look down and pause for a moment before activating a gesture, and then readjust their head to the normal rotation when finished.

WalkTo. In order to prevent our avatars from running into other people and objects in the world, this common behavior was used to add a spatial buffer between the avatar and others in the world. Our other behaviors could call upon this behavior and give it a target location of an avatar, object, or specific location, and in turn this behavior would calculate a small buffer distance and then directly move the avatar to this newly calculated location.

TakeDrink/RemoveDrink. The action of taking a drink, holding it in their hand, and then eventually removing it after a period of time was observed for all roles in our observational study. We created these common behaviors to mimic these observed actions.

5.3.2 Regular

The regular SCR avatar was developed around a set of behaviors focused on interacting with a specific social group of other avatars in *Second Life*, based on interactions during the observational study. Upon initial execution of the agent, the members of the regular's social group consists of the two other regular avatar characters, Terry Rushwald and Ash Kanashimi. New members can be added to this group through a greeting ritual with the avatar, or automatically after the non-member spends enough time within a close proximity of the regular avatar or other regulars.

ChatCircle. The chat circle behavior subtree involves two modes - call and response, and idle/bored actions. If the avatar detects various gestures in the group, such as laughing, pointing, or dancing, the avatar will react with a complementary gesture. If no gestures occur within the social group in 30 seconds, the regular avatar will activate a gesture conveying boredom. Upon the next instance of an idle period, the regular may choose to activate the GetDrink or StageDance behaviors, and then return to the ChatCircle behavior set when finished.

DanceCircle. The DanceCircle behavior subtree involves constant periodic activation of dancing gestures. The regular may copy the dances of others in his or her social group, or autonomously select a dance from the known set. The DanceCircle subtree can be activated when two or more members of the social group begin dancing while in the ChatCircle subtree.

SoloDance. The SoloDance subtree is activated when either the avatar executing the behavior has danced and then receives a clapping gesture from another, or when two members of the social group clap at another member who is dancing. In the former situation, the avatar then continues dancing if claps aimed at the avatar continue to occur, or until a fixed time limit. In the latter situation, the avatar will clap at the dancer and watch, stopping when the dancer decides to stop dancing.

StageDance. The StageDance subtree can be activated from an idle/bored state in the ChatCircle subtree. In this set of behaviors, the regular will walk to the dance floor and begin performing random dance gestures, in an attempt to encourage others to dance with him or her. If other social group members approach and dance with the avatar, then the StageDance will continue to occur. Otherwise, the StageDance will end after a fixed amount of time.

GetDrink. The GetDrink subtree is another behavior subtree that can be activated by a ChatCircle idle/bored state. The avatar will approach a waiter nearby, or the bar otherwise, and then select the bartender object and choose a drink to carry. He or she will then return to the social group and the ChatCircle behavior subtree.

Greeting. If a non-group-member approaches the avatar and initiates the 'Wave' gesture in his or her direction, then our avatar will respond positively and add that new non-member to the list of social group members.

TimeinGroupMembership/TimeoutGroupMembership. In the observational study, often a non-group-member would approach the regular group and never initiate any greeting ritual, instead choosing to immediately dance or attempt to interact with various members. Over time, the non-member would become a natural group member. To handle this type of situation, the regular avatar automatically adds new members to the social group if they are within close proximity of any regular or the avatar for longer than 45 seconds. Similarly, if a member is not within proximity of any regular or the avatar for 30 seconds, then the person is automatically removed from the group member list.

Below is a diagram showing the behavior tree for the regular avatar (see Figure 33). In this image, and in the similar tree diagrams for the waiter and spring breaker, blue nodes are behaviors unique to the role they are depicting, green nodes are reused in various locations within the same role, and orange nodes are global behaviors used by all roles. In all of these diagrams, due to space constraints, we group together multiple variations of the same behavior and don't show minor memory management and utility behaviors. As a result, the actual behavior tree described in the source code is much larger.

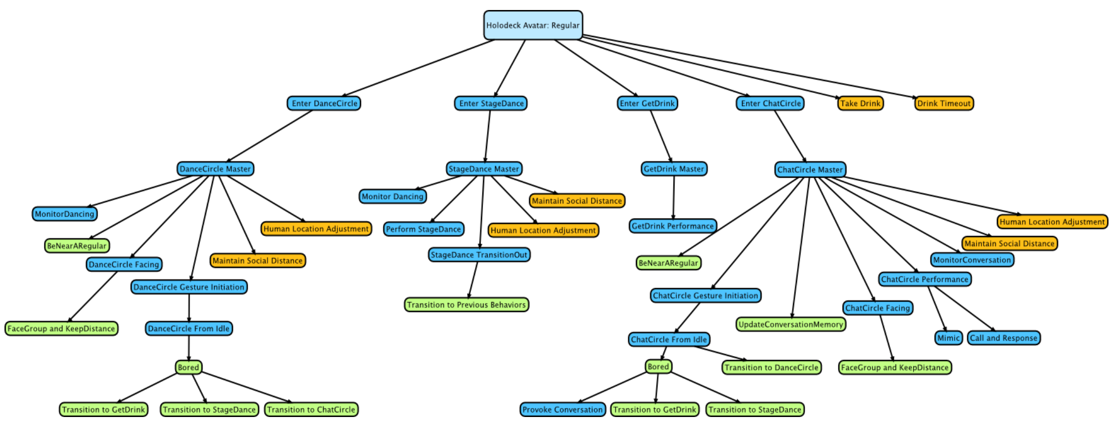


Figure 33 - The regular avatar's behavior tree. The blue nodes are behaviors unique to this avatar role, green nodes are intra-role common behaviors reused in multiple places within a single role, and orange nodes are inter-role global behaviors reused between multiple roles.

5.3.3 Spring Breaker

The spring breaker avatar contains behaviors that are focused around constant positive interactions with various other persons in the environment. These positive interactions include dancing, giving or receiving drinks, and various positive gestures such as 'Yes', clapping, and laughing. In our analysis of observational study videos, the spring breaker confederate would constantly move between persons in the bar, and be very socially aggressive. Because of this, our primary behavior trees focus around finding new partners for dancing, or initiating autonomous dancing in order to encourage others to join him or her on stage.

FindDancePartner. This behavior tree is the primary behavior mode for the spring breaker. He or she will locate a potential dance partner, choosing various roles based on their potential for a positive response. For example, the spring breaker will not likely choose to try to dance with the waiter/waitress or wallflower, because those roles are less likely to respond positively. Once a potential partner is selected, the spring breaker approaches the target and may offer a drink to him or her. If the target responds positively (by dancing or gesturing in a positive manner), then the avatar will attempt to initiate dancing. If the target responds with dancing, then the two will continue dancing until the target leaves or stops dancing, or the spring breaker becomes bored and chooses a new partner. If the spring breaker is offered a drink or approached by one person while not currently dancing with another, then he or she will automatically select that approaching avatar as the target.

StageDance (spring breaker version). Occasionally, during the process of finding a dance partner, the spring breaker will choose to go to the stage in the bar and dance, attempting to encourage others to do the same. This behavior is a modified version of the regular behavior, but with a longer dance time, closer interpersonal proximity, and a faster selection of dances in order to more closely match the personality of the spring breaker role. If no person decides to dance with the spring breaker, he will eventually become bored and return to the task of finding a new partner.

GetDrink (spring breaker version). The spring breaker will actively attempt to maintain the process of drinking in between partner selection. This behavior subtree is a modified version of the regular subtree, with modified timings, interpersonal distances, and gesture selections customized to the personality of the spring breaker role.

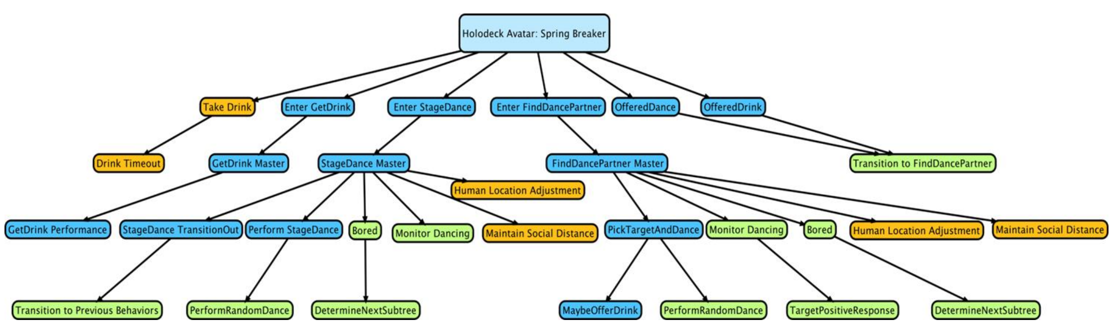


Figure 34 – The spring breaker SCR avatar behavior tree.

5.3.4 Waiter/Waitress

In our observational study, we observed that the waiter/waitress roles consisted primarily of two physically-based modes of interaction - roaming about the bar and dance floor area attending to or watching customers, and staying behind the bar to attend to those close to that area.

Roam. In the Roam behavior subtree, the waiter/waitress avatar moves between many locations around the bar. In each of these locations, he or she will look at various customers, occasionally clap at dances, or laugh at dances or other gestures. The waiter/waitress will also focus attention at any people in the environment that are standing close, because those may be customers that want a drink. Occasionally, the avatar will select a customer nearby and approach him or her, transiting to the ServiceCustomer behavior subtree.

BehindBar. When the waiter/waitress is not servicing a customer, he or she may choose to move to the area behind the bar. From this location the avatar will observe the dance floor, or focus attention on any customers close to the bar, gesturing positivity to encourage drink selections.

ServiceCustomer. If the waiter/waitress is waved at or closely approached by a customer while roaming, or if he or she chooses to approach a customer while in the Roam subtree, the ServiceCustomer subtree will activate. If the waiter/waitress is acting as the initiator, the waiter/waitress will give a drink to the selected customer, then gesture in a friendly manner, wait for a response, and then return to the roaming behavior. If approached, the waiter will gesture positively and wait for the customer to select a drink from the bartender object in his or her hand. The waiter may also (rarely) choose to directly offer a drink to the customer as well.

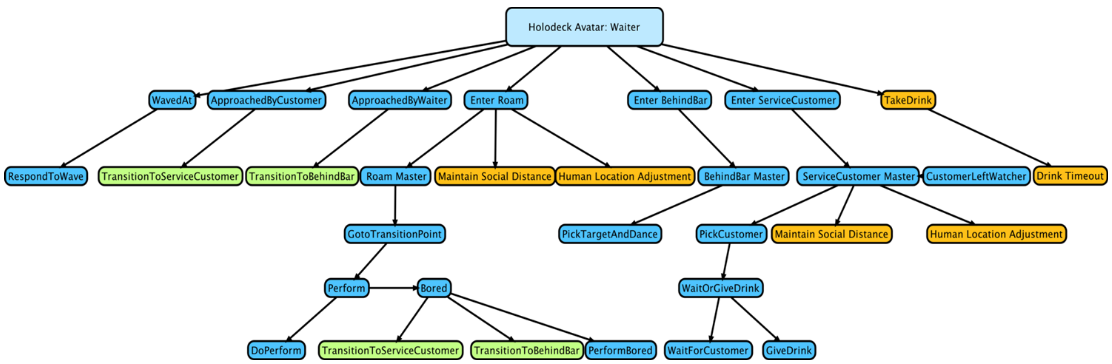


Figure 35 - Waiter Avatar Behavior Tree

6 Full Study

As a trial prior to our full study, we conducted a pilot study, consisting of 5 participants (4 male, 1 female) to assess how the procedure and tasking from our previous observational study needed to be tweaked given the introduction of SCR avatars to the virtual bar. We did not conduct full statistical analysis of the pilot study; the main goal was to iron out procedures and get feedback from participants. As planned, we added our newly implemented SCR avatar personalities for the pilot: a regular, a spring breaker, and a waiter/waitress, and given the participant feedback and our analysis in the observational study, we added the generic bot role and removed the human wallflower role, for a total of

11 avatars comprised of five human confederate avatars, three SCR avatars and three bots. Pilot testing showed that it was confusing to have several waiters in the small space, and that overall there were too many other avatars to ‘get to know’ in a short period of time. As a result, we further refined the roles for our full study by removing two waiter/waitress avatars.

Our full study, with the additional SCR avatar roles and feedback from both our observational and pilot studies, consisted of: 3 human confederate controlled avatars (2 regulars and 1 spring breaker), 3 SCR avatars (1 regular 1 spring breaker, and 1 waiter/waitress), and 2 bot controlled avatars (1 wallflower and 1 generic). We had a total of 24 participants (4 females and 20 males) over a span of 3 sessions. Additional changes to the full study included a simplified and compacted participant task list, overhead video capture view of the scenario to supplement participant view capture, and the addition of a time limit as an independent variable. From our previous analyses, we had identified that time in the bar could have an impact on a subject’s interactions with other avatars, and so we set explicit time limits for the amount of time participants had in the virtual bar - half of the participants had a 12 minute time limit to complete the task list, while the other half had 24 minutes.

Table 25 - Candidate metrics for the Holodeck project's full study based on results from the observational study.

Metric	Description
Gaze Angle	The angle an avatar faces relative to the direction towards a target. It ranges from 0 (facing target) to 180 (turned directly away from target).
Gaze Sum	Sum of the gaze angles between two avatars; it ranges from 0 (two avatars facing each other) to 360 (two avatars looking completely away from each other).
Interpersonal Distance (IPD)	Pair-wise distance between avatars.
Space	Categorized as public, social, personal, or intimate.
Gesture Target	The average number of gestures targeted at an avatar.
Co-presence	The extent to which participants reported behaving and responding as if the avatars were real.

6.1 Measures

A variety of social interaction dimensions exist which can be used to evaluate the interactions in our experiments. For example, Blascovich et. al.'s threshold model of social influence (Blascovich et al. 2002) considers social presence and behavioral realism when accounting for interactions between avatars in virtual world settings. Under this framework, categories of observable behaviors can be used to predict the level of social influence a person might be experiencing. For example, prior work by Bailenson (Bailenson et al. 2004) has shown that interpersonal distance is one such reliable indicator: people move closer to avatars that have a lower level of realism than those with a higher level of realism. Our observational study results presented a set of viable candidate metrics which showed measurable differences between the human and bot controlled avatars, shown in Table 25. We used these metrics as a potential indicator of realism in our full study.

6.2 RESULTS AND ANALYSIS

In Table 26, we summarize the results of the within group analysis of variance (ANOVA) for the following measures:

- Gaze Angle: from the participant to the various avatars
- Gaze Sum: Two avatars facing towards each other
- Interpersonal Distance: pair wise distance between avatars
- Gesture Target Towards Avatar
- Co-presence

Analysis was conducted for the main effects of:

- Avatar Type (3) – Human, SCR avatar, and Bot
- Avatar Role (8) – looking at the 8 avatars individually

Table 26 - Statistics summaries of the measure for the Holodeck project's full study.

Measure	Effect	df	F	Sig.	Power
Gaze Angle	Type	1.61	502.62	.000	1.00
	Role	4.31	277.23	.000	1.00
Gaze Sum	Type	1.84	1009.04	.000	1.00
	Role	6.05	728.97	.000	1.00
IPD	Type	1.58	1968.22	.000	1.00
	Role	4.06	2410.23	.000	1.00
Gesture Target	Type	1.77	10.71	.000	.97
	Role	3.68	7.51	.000	.99
Co-Presence	Type	1.97	2.76	.075	.51
	Role	4.74	10.54	.000	1.00

The effect column of Table 26 shows what aspects of the avatars the data in the row is relative to. It is either the type of avatar (human, bot, or SCR controlled) or the roles (such as Regular). The remaining rows show the results of the ANOVA analysis. The third and forth columns show the results of the F test[41]. The third column shows the estimated degrees of freedom in the F test. The fourth column shows the F value or the ration of between-group to within-group variance. The fifth and sixth columns report significance values at $p < .001$ with an observed power > .95, unless otherwise stated. Follow-up within subject contrasts were conducted using pairwise comparisons. Significance for pairwise comparisons are reported at a p <.001 level of significance.

A 4 (space) x 3 (avatar type) repeated measures analysis was conducted for the Space metric to look at possible interaction effects of social space and avatar type. These results are not included in Table 26.

[41] F tests build models of a data set (commonly using the least squares algorithm) and compare the variance of the data to the model. This allows the data to be tested against the model for non-normality. The *F* tests shown above are reported for the within group effects include the Greenhouse-Geisser correction when necessary to protect against possible violation of the sphericity assumption.

6.3 Gaze Angle

From the captured data, we calculated the gaze angles (derived from previous work by Yee et al) between the participant avatar and all other avatars (Nick Yee et al. 2007). A lower gaze angle means that the participant was more often looking more directly at an avatar.

Figure 36 and Figure 37 show the results distributed by avatar type.

This figure illustrates that participants maintained the smallest gaze angle with human avatars, followed by SCR avatars, and the largest with bots. The main effect of avatar type on gaze angle was significant (see Table 26). In addition, follow-up within subject contrasts showed that all pairwise comparisons were significant. We can see here that SCR avatars performed closer to human controlled avatars than traditional bots maintaining a more direct gaze at our participant, similar to our human confederates in the scenario. This move towards the human end of the spectrum benefits our target of SCR.

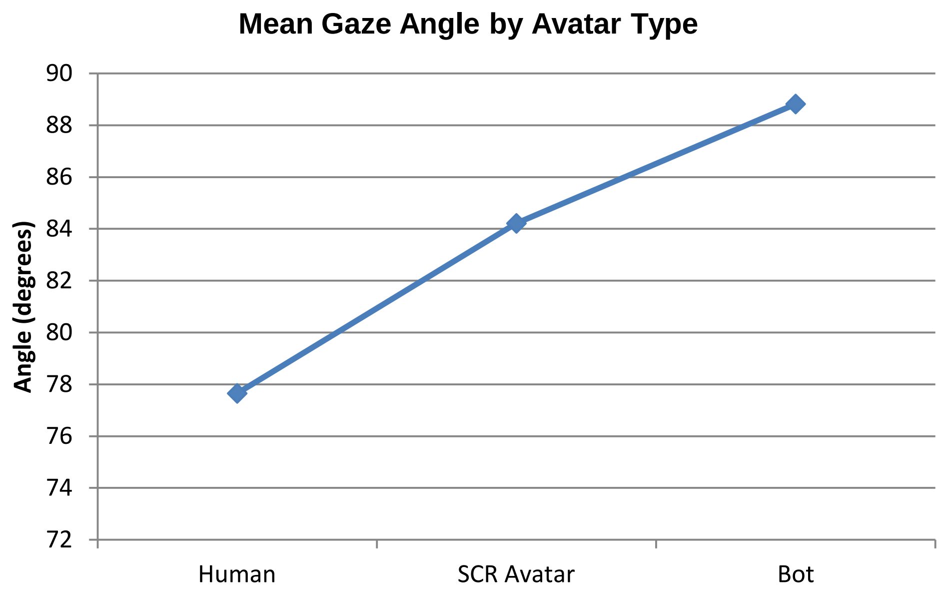


Figure 36 - Gaze Angle means from the Participant to Avatars by Type

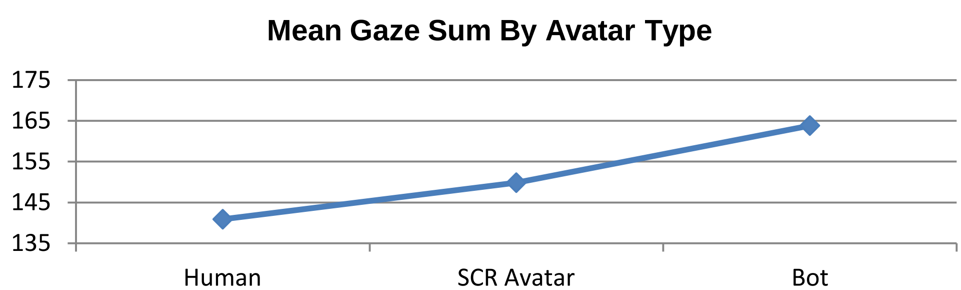


Figure 37 - Gaze Sum means between the Participant and Avatars by Type

6.4 Gaze Sum

The Gaze sum is a combination of two gaze angles between two avatars, as proposed by Yee et al (Nick Yee et al. 2007). As mentioned in a previous section, this measurement quantifies how much two avatars are looking at each other, with zero degrees representing two avatars looking directly at each other, and 360 degrees corresponding to two avatars looking directly away from each other. In this analysis, we measured the mean gaze sum of each avatar through runs of our studies by combining the gaze angles between each avatar type and the participant towards that specific avatar, and grouped these sums by type.

Figure 37 illustrates a significant effect for avatar type on gaze sum. Follow-up within subject contrasts showed that all pairwise comparisons were significant. Like with our gaze angle measurements, the mean gaze sum similarly shows that our SCR avatars place between the performance of a human and bot, approaching the target of human-level behavior.

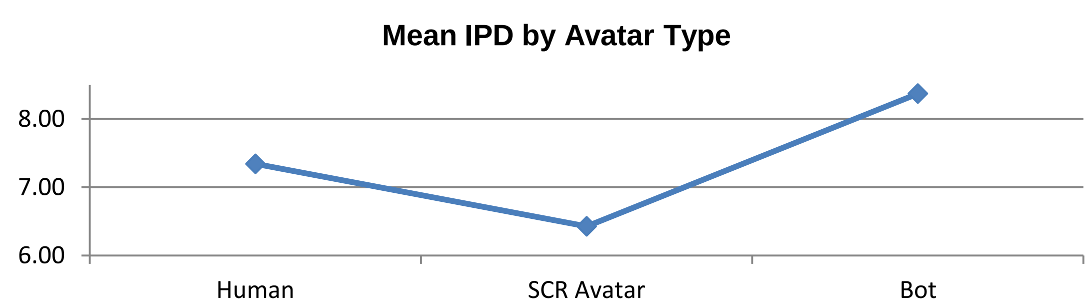


Figure 38 - Interpersonal Distance means by Avatar Type

6.5 Interpersonal Distance (IPD)

Interpersonal distance (or IPD) from the participant was calculated (in meters) from the positional data that was logged for all avatars. Both avatar type and avatar role had a significant effect for avatar type on interpersonal distance. Figure 39 shows that our SCR avatars tended to have a smaller distance to the participant compared to bots. This shows that the players actively stayed closer to the SCR avatars than the bot avatars. This view of IPD also shows that the SCR avatars were staying closer to other avatars than bots were. A strong example of this is in Figure 39 which shows that our spring breaker almost always was closer to the participant than any human or bot, due to particularly aggressive social behaviors that constantly aimed to dance with all of the patrons of our virtual bar. In future work, in order to blend in with the human confederate avatars, this particularly outstanding behavior would be modified to more closely match the distances achieved by the human spring breaker.

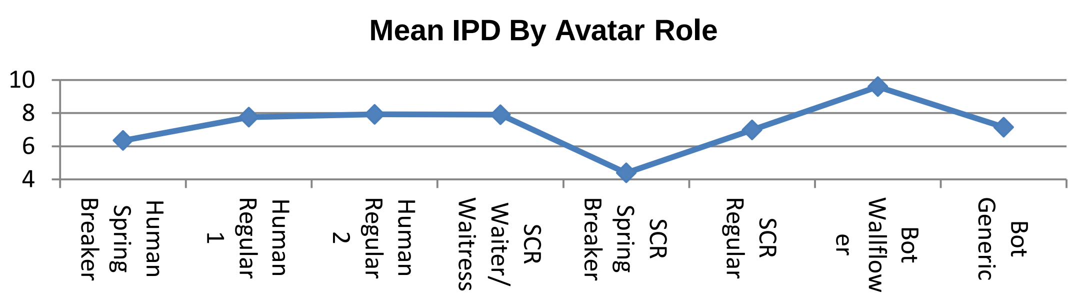


Figure 39 - Interpersonal Distance means by Avatar Role

6.6 Space

Space was calculated using the personal reaction bubbles developed from Hall (Hall 1966). The interpersonal distance measures were then grouped into four bins: Intimate space (< 0.45 m), Personal Space (< 1.2 m), Social Space (< 3.6m), and Public Space (> 3.6 meters).

A 4x3 (space categories by avatar type) repeated measures analysis demonstrated a significant interaction effect for avatar type on space (see Table 26). Participants spent most of their time in public space with all avatars, but this was highest for Bots. Figure 40 shows that our SCR avatars tended to stay at the same distances from our participants for the same percentage of time as our human confederates.

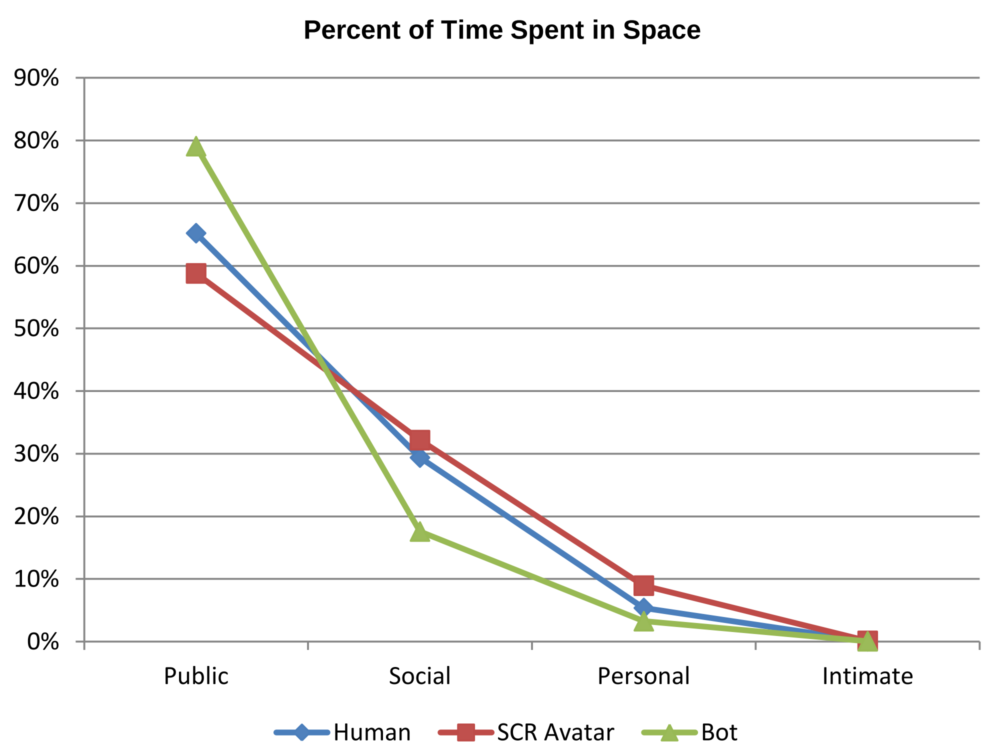


Figure 40 - Percent of time spent in various space categories by avatar Type.

6.7 Gesture Target

Gesture targets towards the participant to other avatars were calculated by finding the nearest avatar facing the participant when the participant was using a gesture. The avatar that was facing the participant and closest to him or her was counted as the target for that particular gesture.

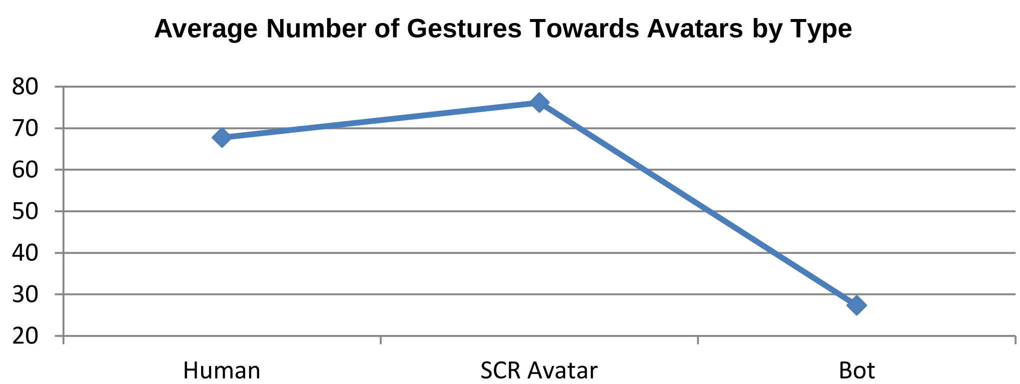


Figure 41 - Average number of gestures toward avatars by types.

Both avatar type and avatar role had a significant effect on the average number of gestures used by participants (Table 26). Follow-up within subject contrasts showed that there was a significant difference between human and bot, and SCR avatar and bot, but the difference between humans and SCR avatars was not significant (Figure 41). A repeated measures analysis demonstrated a significant effect for avatar role on gestures (Table 26).

Participants gestured significantly more to the SCR spring breaker than any other avatar (Figure 39) - a result of this avatar's aggressive social behaviors towards the participant and other avatars in the bar.

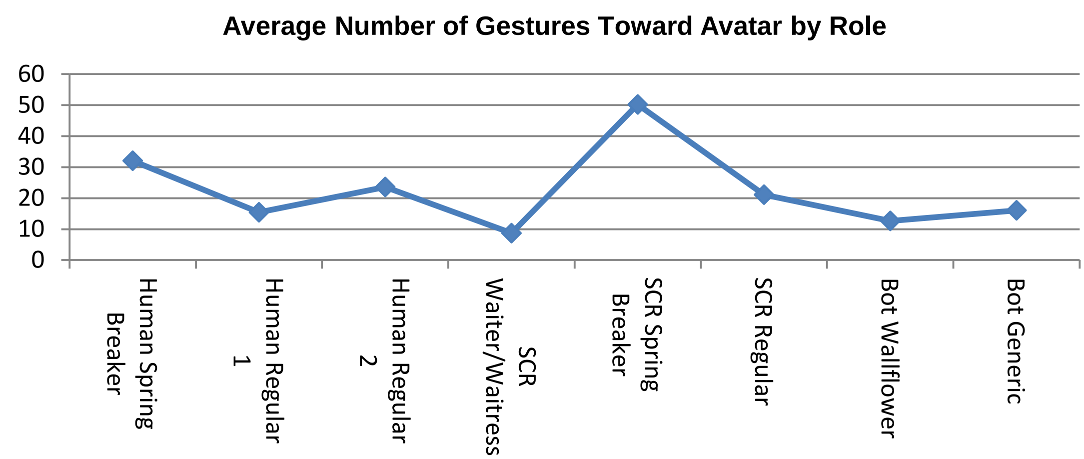


Figure 42- Average number of gestures towards avatars by role.

6.8 Co-Presence

Co-presence refers to the participants' sense of being with another person in the virtual bar. The co-presence questionnaire was adapted from the Slater Co-Presence Questionnaire (Inman, Wright, and Hartman 2010). Participants used a 7-point numerical scale (1. not at all to 7. a great deal) to respond to seventeen items (such as "I had a sense of being with the other person…" or "The experience seems to me more like interacting with a person…"). Responses to the items were used to compute co-presence mean, which was the average rating across all of the items.

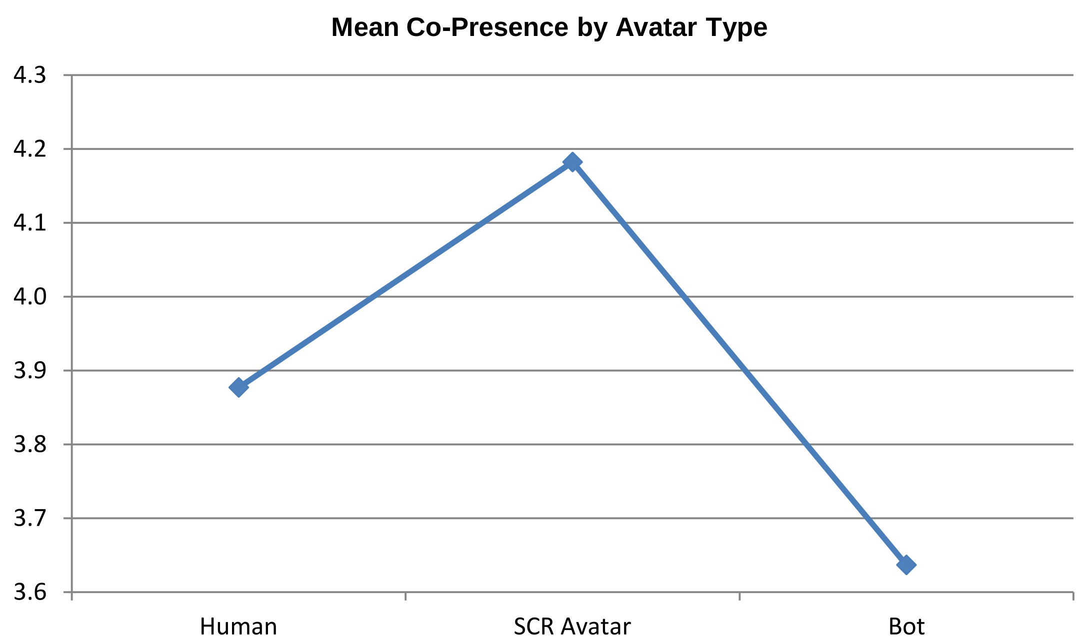


Figure 43 - Co-presence means by avatar type.

Although the effect of avatar type on co-presence mean was not significant, co-presence was highest for the SCR avatars (M=4.18), followed by human (M=3.88), and lowest for bots (M=3.64) (Figure 43). Our analysis also shows a significant effect of avatar role on co-presence (Table 26). While not all of our prototype avatars score significantly better than other roles, the sense of co-presence was significantly higher for the SCR spring breaker (Figure 44), showing great potential for the for future development of other roles that learn from the lessons of our spring breaker's design.

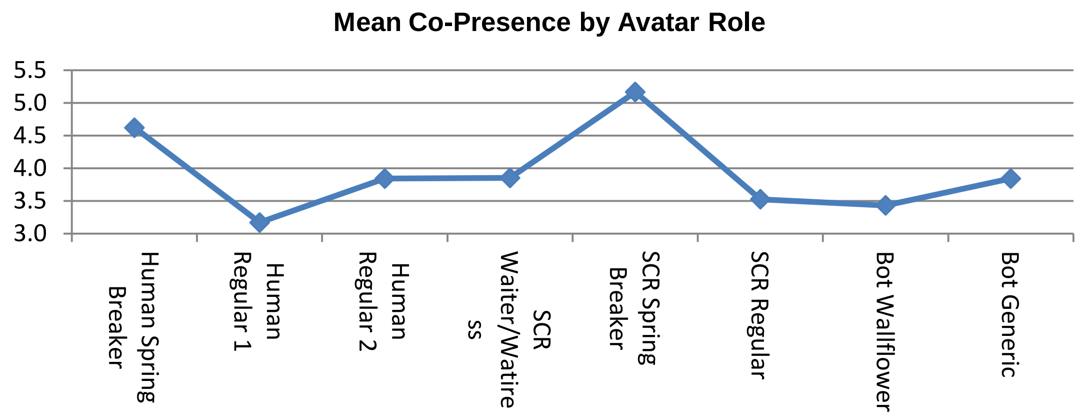


Figure 44 - Co-presence means by avatar role.

6.9 Debriefing and Comments

During the participant debriefing sessions, several of the participants indicated that they would have really like to be able to use chat in the virtual bar in order to complete the assigned tasks. Participants also felt that chat would have made their experience much more realistic. Overall, participants found the experience “cool”, “fun”, and “interesting”. However, they felt it was hard to understand the reactions of the other avatars. They also had trouble knowing whether or not other avatars “received/acknowledged” gestures intended for them.

Participants thought the other avatars were realistic with distinguishable personalities. However, many felt that the virtual bar was not crowded enough, and would have liked a live band, more people, etc.

The following are sample participant comments from our debriefing sessions (note that Pat, Harper, and Alec were SCR avatars in our full study):

- "...Pat, Ashe and Terry were a tight knit group. Quinn was that awkward newbie. Devin, was the one used to the world and was just having fun doing her own thing. Harper was fun, but a little outrageous. Terry...that smug b****. Parker, that goth non-conformist. Pat...Pat was Pat. He had no characteristics, but then again he was in a tight knit group"
- ".. one point I was wondering if someone was actually playing the people. Their responses to my actions and movements made the game pretty fun. ... Ashe was the social one, but was distant to the new guy ...Terry was the same, and was in the same group as her. Devin was the dancer, she just danced, usually by herself. Alec was the guy who you had to get to know to have him like you. Quinn was the person who just right off the bat gave you a drink, and Harper was the person who immediately started dancing with you. Add in Pat who refuses to dance with other people, and Parker who just refused to even look at me, and you have an entire cast of people who you can find in high school. However, they were believable characters, and I did come up with all this after only playing a few minutes, even if I'm off I still have a firm idea of what kind of character they are..."

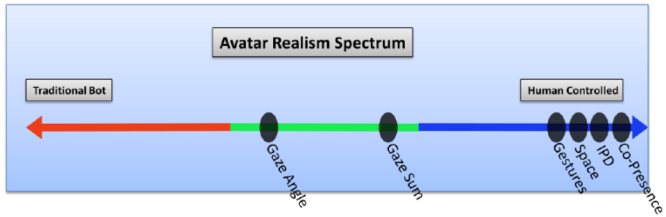


Figure 45 - An illustration of relative performance of the SCR avatars to human and bot controlled avatars for each metric.

7 Discussion

In the observational study, we were able to establish both objective (gaze angle and sum, IPD, gesture target, and space categorization) and subjective (co-presence) baseline metrics as strong indicators of agency in comparing human controlled and bot controlled avatars. The goal of the full study was to evaluate our prototype SCR avatar with respect to those established candidate metrics, and see where these metrics fall on the "Avatar Realism Spectrum" for the SCR avatars (as previously seen in Figure 31).

Our hope was that our SCR avatars would score closer to human avatars than traditional bot avatars. As you can see from our results, we did just this: our prototype avatars scored closer to humans than to bots with statistical significance for all analyses in our metrics but co-presence, and our SCR spring breaker showed a significantly higher co-presence for his or her role than any other character in our scenario. Figure 45 illustrates relative performance for each metric using an ad hoc summary comparison.

7.1 Duration Effects

In general, our longer 24 minute study runs showed significant effects on gaze angles, gaze sums and interpersonal distances with our SCR avatars, with higher values on all of these measures. This effect on duration likely shows that as participants spent longer periods in our virtual bar, they spent more time with our avatars and lost interest with these interactions, stressing the capabilities of our SCR-focused behaviors. Extended direct interaction time, even when it does not include language interaction, is a condition that begins to move out of the realm of "supporting character." Future work would involve examining where and how the behaviors of our SCR avatars cause participants to start losing interest (perhaps suspecting that they are computer controlled). These longer runs showed no significant effects on gesture target measures, and the lack of significance for our co-presence measures for all participants did not warrant analysis based on duration.

7.2 Gender Effects

In general, females in our study tended to treat our SCR avatars more like human controlled avatars, with lower gaze angles and sums, closer interpersonal distances, and a higher sense of co-presence. However, since we did not have enough female participants to demonstrate significant effects, we leave a full analysis of these gender effects to future work.

7.3 Limitations and Future Work

Our work here shows that, given our baseline metrics, SCR avatars can perform comparably to human-controlled avatars in performing supporting character roles, and almost always significantly better than traditional scripted bots. Future work should examine the effects of gender on the perception of co-presence and realism that were hinted at in this work. The behavioral repertoire of SCR avatars should be expanded to support a wider palette of physical interactions. While *Second Life* provides a large number of gestures and the ability to add additional user-provided animations, it contains lower fidelity and primitive animation and interaction functionality compared to many contemporary game engines. The addition of full facial expressions, higher resolution animation skeletons and models, accurate and realistic physical collision, and more nuanced animations with smooth transitions and blending would allow our agent to explore a much larger pallet of social interaction, and allow our participants to more easily perform tasks by accurately gauging avatar responses and subtle emotional states. These interactions could include subtle changes in posture and facial expression, direct touches or hugs, and layered emotional responses that can be expressed differently based on their location in the virtual body. Finally, future work should explore the possibility for limited language capability of an agent targeting SCR. Establishing metrics that evaluate the linguistic skills of a character for the purposes of SCR will be critical in this future work.

7.4 Conclusion

In this chapter, we have established SCR both conceptually and experimentally. We first proposed the concept of SCR and distinguished it from believability and Turing Test realism, and created an experimental scenario that allows human participants to interact with mixed groups of humans, bots and SCR avatars. By analyzing the data from the first of these experiments, we then established a set of candidate metrics to measure the interaction amongst humans and computer controlled avatars. Finally, we showed that our prototype SCR avatars significantly outperformed traditional scripted bots and approached the interaction realism of the human confederates playing supporting characters. This chapter provides both a method of defining and measuring SCR, and a demonstration that it is technically possible to approach SCR with contemporary technology.

The *Holodeck* project addresses the aspects of performing social interaction in real time that are not addressed by *CiF.* Specifically, real-time social interaction, non-verbal performance, social proximics, and rituals are exhibited by SCR avatars and are evaluated. The technology, representation, and experimental methodology used in this chapter provide a solid foundation for enabling social interaction in real time with *CiF.*

Chapter 7

Authoring Content for Playable Social Models

This chapter has two goals. The first is to provide a general description of how to author content for *CiF*. As creating this type of content requires the author to think at the intersection of rule building, storyworld creation, and ethnographic analysis, there is no simple step-by-step tutorial to guide a potential author. Instead, the general flow, useful patterns, and general advice to help guide the authoring process is presented. This description is reinforced by a discussion based in sociology about why ethnographic analysis is a good fit for authoring interactive experiences.

The remainder of the chapter consists of case studies of authoring in this style. First is the initial study of *Sex and the City* that helped refine and develop the ideas of social exchange and character-based performances. The second case study is on authoring with *CiF* for *Prom Week.* Even though the *Prom Week* case study is the most complete and refined authoring experiences with *CiF*, the AI system and the game were being iteratively designed throughout the development of *Prom Week*. In the discussion of authoring for *Prom Week*, parts of the description include iteration on the method itself. The final case study shows how the *SCR avatar*s were created through the analysis of data taken from human role players.

1 Method for Authoring a Storyworld with *CiF*

The general method for authoring a storyworld in *CiF* consists of refining the domain of the storyworld through analysis of the story space. As will be discussed in both this section and the authoring examples later in the chapter, this method is an iterative process of refinement; each step solidifies the domain and makes the analysis more efficient. This cycle needs to be bootstrapped through having a general area or domain for the basis of the storyworld. In the examples later in the chapter, these domains can consist of a range of foundations; something as vague as a typified space (such as a virtual bar) with culturally associated normative social roles or something more concrete like specific pre-existing media sources (such as films or books). These serve to both narrow down the range of possible social spaces (from no constraints down to a high school or bar) and the scope of the interactions (cutting out things like long cultural patterns and the mundane, low drama interactions in favor of high-drama high school hallway politics).

With a social space in mind, the most general aspects of that space can be described and serve as a foundation for more specific details. As relationships are primary social factors in this method, the first step consists of determining the salient relationships and social networks of the social space. This provides a basis for authoring the social exchanges by defining a broad space for social change. With a number of social exchanges (associated with basic social change) in mind, character traits and statuses that are germane to the social exchanges and relationships can be defined. With a working set of statuses, character traits, relationships, and social networks of the story world defined, a basic social state space is available for building performances.

With the social state space in mind, outlines of performances can be made for the social exchanges. At this point, it is often advantageous to review the source material (*Mean Girls* and *Twilight* in the case of *Prom Week*) for specific performances of social exchanges. At this point in the authoring process, the nuances of the bindings of performance, social state change, and normal social behavior come in to new focus. The space of what can be authored opens up drastically; authoring performances exposes the social considerations of the characters. This provides the raw material for creating influence rules which can be grouped into microtheories. For *Prom Week*, we found grouping microtheories based on the authoring hierarchy worked well. We separated microtheories based on the predicates they included in a hierarchical order: first were influence rules that contained relationships then statuses, traits, CKB, and history references.

After social exchanges are associated with some performances and with microtheories roughly approximating social norms, the time is right to create characters. As characters are one of the lightest areas of representation in *CiF*, they are fairly simple to construct;

simply assign traits and names. With the addition of characters[42], *CiF* is now able to simulate your social world. From this point on, the authoring process involves filling out the social space and refining the authored content. When the experience is playable, the parts of the domain that are not modeled in *CiF* become apparent; characters will ignore social states that players expect them to react to. User testing is very important in debugging and tuning the modeled social knowledge as players with a fresh perspective on the story world will be quick to notice breaches in normal social behavior. The authoring paradigm can then bounce from being driven by the author's ideas about the domain to the game experience of players.

With this type of authoring, missing spaces of authored content higher in the importance hierarchy (such as social exchanges or even relationships) is to be expected. It is critical to step back and holistically assess the story space; when authoring performances for cheating in *Prom Week*, we uncovered the need for several new statuses ("homewrecker", "homewrecked", and "cheated on by") that allowed the construction of new microtheories to catch the intricacies of expected responses to characters cheating.

[42] Depending on the preferences and style of the authors, creating characters may need to happen earlier in the story world building process. As *Prom Week* was meant to be overtly identified as a simulation, the characters were finalized late in the design process. If you are creating an experience that is very character-driven or you are modeling a specific story world (such as a direct modeling of a pre-existing storyworld), characters may be more crucial to your experience and should be built much earlier in the authoring process and constantly thought of through all states of authoring.

2 Social Exchanges in *Sex and the City*

This analysis transformed our understanding of how to represent social exchanges and their associated social reasoning. The ideas matured from a vague form to the foundation for first prototypes of *CiF* and *Prom Week*. Our goal was to identify and explicitly represent dramatically interesting social exchanges that can serve as a basis for our playable model – the model itself was being authored by this work. The importance of relationships and statuses being used in tandem with internal psychological states and moods became apparent. The resolution (or representative scope) of social exchanges determined what is useful and appropriate for procedurally enacting highly dramatic inter-character performances. All of these design insights were critical in the design, deployment, implementation, and authoring of *Prom Week*.

As a first exploration into using ethnomethodology to uncover knowledge useful to constructing AI-driven characters, I applied dramaturgical analysis to interpret the situations and the actions taken by characters in the dramatic setting of the HBO show *Sex and the City* (Star 1998). We focused on interpersonal dramas such as *Sex and the City* under the assumption that, for such dramas, the screenwriter is implicitly structuring the social interactions as dramatically interesting social exchanges. This makes it easier for us to isolate compelling social exchanges than it would be performing a similar analysis on naturalistic interactions from everyday life. Individual interactions were viewed in terms of the roles taken by the participants, the setting of the interaction, teams composed of the participants, who comprised the audience, etc. With the interactions represented in this dramaturgical way, patterns became easier to distinguish within the drama.

As previously mentioned, dramaturgical analysis provides a method to capture these interaction patterns while keeping the normal social interaction and the context of the pattern intact. To better understand how this is accomplished, the process is explored in the following example from an analysis of a scene from *Sex and the City* (King 2001). The setting of the performance is a wedding engagement party held in an expensive apartment in Manhattan. The audience consists of several dozen upper-middle class married individuals who are all in some way socially connected to the newly-engaged couple. The props present are objects typically found at celebrations: champagne glasses, tables, chairs, presents, etc.

The cast consists of two single friends, Miranda and Carrie, and a group of several female acquaintances, all of whom are married. Carrie and Miranda have two very distinct personalities. Carrie is an outgoing person who tends to directly face situations one at a time and is very focused on the role she is playing. Miranda, while being focused like Carrie, prioritizes the avoidance of bad things over proactively seeking her goals.

The scene begins and plays out in the following sequence of events. Carrie, Miranda, and the group of women are engaging in conversation when the topic of relationships is brought up. The group of women discusses their current relationships. Eventually, the focus is placed on Carrie and she is asked about her relationship status. She states that she is single and content, then passes the conversation back to the group. They then proceed to similarly ask Miranda about her relationship status. Miranda responds by going into a round of self deprecating jokes about her being single. She then excuses herself from the scene and exits the stage with Carrie; the scene ends. The following is a transcript of the scene:

Group Leader: From the moment we met I knew it was perfect. I we were meant to be together. I could stop looking; I had found him!

Carrie and Miranda: Congratulations!

Group Leader: What about you guys?

Carrie: Oh, well I am not dating anyone.

Group Leader: Oh. And what about you, Miranda? Seeing anyone special?

Miranda: Actually no. But I am a whole bunch of un-special guys.

Group laughs.

Miranda: That's one of the requirements to date me. Are you special? Sorry, move along. But if you have any un-special friends, could you give them my number?

Group laughs and Carrie stares at Miranda.

Miranda: I'm serious! Do any of you have a completely unremarkable friend or maybe a houseplant I can go to dinner with on Saturday night?

Group laughs and Group Leader turns to Carrie.

Group Leader: She's hilarious!

There are many possible ways to interpret this scene as social exchanges. The ritualistic way Miranda makes a joke and the group laughs is a social exchange. The entire scene is part of a long term social exchange, or life game, between Miranda and Carrie about Miranda's self esteem. Because the scene takes place in a setting loaded with dramatic possibilities, I set constraints on the analysis based on the storyworld I wanted to create at the time, which was creating an interactive narrative in the style of *Façade*. With these constraints in mind, I was analyzing the scene to find short term, small group interactions. As *Sex and the City* is a fictional television series, there were strong cues to finding appropriate social exchanges. The cinematography of the scene, including characters' locations in the scene, camera shots and camera cuts, helped frame the analysis by cutting out the background activity seen in everyday social interactions.

Considering the constraints, a clear social exchange is present in the previous example. The actions taken by Carrie, Miranda, and the group of women result in a social state change, altering what the group thinks of those who do not share a vital part of the group's identity (the identity of being in a relationship). Two characters, Miranda and Carrie, have the goal of using the actions in the scene to manage the impression that the group of women will form of them. The cast of characters involved and how this social We call this social exchange "not like the others."

This example illustrates an important concept related to dramaturgical analysis: impression management. Impression management is a goal-directed attempt at influencing the perceptions that are formed by others, which can be performed either consciously or unconsciously. It is not necessary for the subject of the managed impression to be the one performing the managing; the impressions can be of another person, event, or arbitrary object. When one manages the impressions of oneself, it is called self-presentation. Impressions are managed by regulating social information and interactions. Both Miranda and Carrie had the goal of self-presentation with regards to the group of women in the social exchange. This specific pattern of impression management found in "not like the others" is important; the social outcome for each role is what makes the social exchange unique. If another social exchange was found with identical roles and social ramifications but had a different performance, either the concept of "not like the others" would expand to include the new performance or another way to perform the social exchange would be recorded.

The context of the episode around this scene from *Sex and the City* shows the compositional nature of performances in the dramaturgical metaphor. Because Carrie and Miranda are friends of the groom-to-be, they are playing a social exchange of supporting their friend's engagement. The example, therefore, is an event in the larger social exchange of supporting a friend. Furthermore, by planning to go to the party together, Carrie and Miranda are presenting mutual support while simultaneously playing the example social exchange.

3 Authoring Patterns of Non-verbal Behavior

To create the patterns of social proximics, gestures, and role-sensitive behaviors exhibited by the *SCR avatar*s, we employed analysis techniques similar to those used in authoring for *Prom Week*. However, instead of using pre-existing media as the material for analysis, we used gameplay videos and automatically logged data from human role players in *Second Life.* The role players took on the roles found in the *Holodeck* project's experimental design: the spring breaker, regular, wallflower, and waiter. By using the resultant performances of the role players acting out typical roles in the bar setting, we had a reflection of the human actors' sense of the normal patterns of social behavior in the space to draw from when authoring the *SCR avatar*s.

The visual analysis provided a general sense of what the human role players were doing in each role in specific contexts. Unlike visual analysis methods that take preprocessed video footage and use algorithms to infer beliefs to create models of behavior (Li et al. 2009), we performed an interpretive visual analysis based on experience participating in the *Holodeck* virtual bar. Additionally, some roles had more tightly constrained and well known performance patterns than others. For example, the waiter is both an occupation and a role in a generic bar setting. Furthermore, the role of a person in a wait staff position must perform to expectations that are generic of wait staff in any venue; at a bar, a French restaurant or at White Castle, there are common expectations of service that must be met to not incur a breach of normal social behavior with customers. The waiter role in our virtual bar setting has all of these generic waiter expectations along with the ones specific to a bar setting. Cleaning the table of abandoned drinks, taking drink orders, explaining what is in specific items, and authoritatively dealing with the abnormal and outrageous behavior of bar goers are all expectation of the role. The following are examples of the visual analysis of the waiter role behaviors that correspond to the mentioned role expectations (the distinct behavior cases are broken up by empty lines and the performances are described by general physical interactions with performance qualifiers like "friendly"):

Participant: walk up
Waiter: wave/be friendly

Waiter: stand behind bar, staring at person closest to bar, acting friendly

Waiter: one waiter behind bar at a time, coordinate with human waiter/waitress

Participant: Dancing at waiter/waitress
Waiter: Kindly decline to dance and get back to regular tasks.

Participant: leaning/sitting doing nothing
Waiter: offer drink, be friendly

Spring breaker: dance next to, be too close
Waiter: pissed off, annoyed
or Waiter: friendly, offer drink

While the waiter role has many well-defined expectations, other roles, such as the spring breaker, were more abstract and were defined by the performances of the role players. The spring breaker role was presented to the role players in a vague manner that allowed for a high degree of interpretation. After looking at videos, this role was less constrained at the level of detailed performance than the waiter. The essence of performance that defined the role was in the general attitude portrayed by the avatar. Energy, near manic socialization, high gesture frequency, and always trying to be near groups of other avatars defined the spring breaker. Also, as seen in the following examples, there was a judgmental element to the spring breaker; if another avatar is not partying, they are not worthy of attention:

Spring breaker: Wave
Participant: ask to dance response
Spring breaker: dance

Spring breaker: dance with regulars
Participant: dance with spring breaker

Spring breaker: dance with whoever else is dancing on stage

Spring breaker: see person dancing on stage
Spring breaker: clap at dancer, get near them

Spring breaker: gives drink
Participant: ignores
Spring breaker: offended,
or Spring breaker: doesn't care, moves on to another person to party with

Spring breaker: walks to participant
Participant: laughs, or mocks
Spring breaker: ignore and keep dancing
or Spring breaker: take offense, be angry
or Spring breaker: take party elsewhere

Participant: leaning or sitting
Spring breaker: force interaction, interfere

Participant: dancing on stage alone
Spring breaker: dance with them

Spring breaker: point and laugh at somebody

As the results attest, this type of analysis of the observational videos and data provided detailed enough information to author agents that performed close to human standard in this virtual world. Specifically, the results of the gathered qualitative and quantitative data shows that the *SCR avatar*s' (who were the beneficiaries of this ethnographic-style of information gathering) behavior was much more like human players than the standard bots found in *Second Life*.

The quantitative data (consisting of avatar proximity and gesture type and frequency) supported the analysis of the videos in ways that had direct impacts on the performance of the *SCR avatar*s. As seen in Observational Study in chapter 6, avatar performance data was gathered in real time as the study progressed. Gestures, avatar locations, and avatar facing are examples of what was gathered. We used this information to tune the behaviors of the *SCR avatar*s in several areas. First, the interpersonal distance between the avatars was correlated with observed behaviors to create a model of social proximics. Also, we performed a similar correlation using character facing (the equivalent of where the avatar is gazing) to give the *SCR avatar*s better gaze etiquette. Finally, we looked at the gesture types and frequency data to refine the gestures of the behavior trees.

While over a slightly different domain, the authoring done to create the *SCR avatar*s provided insight that future versions of playable social models could leverage. By creating behaviors that perform in real time for the non-verbal equivalent of social exchanges, the ground work was laid for a future for *CiF* in virtual worlds with continuous character interaction.

4 Authoring for *Prom Week*

This section chronicles the authoring for and related design issues of creating *Prom Week*'s fictional world. As *Prom Week* is one of the two games (the second is *Mismanor*) that deeply incorporates content created in *CiF* into the gameplay, this section's description of the authoring process and design insight is a necessary compliment to the Method for Authoring a Storyworld with *CiF* section for any future author, designer, or scholar that wishes to explore the creation of content for procedural story spaces. The following topics are covered in this section: the analysis of pre-existing media sources underlying *Prom Week*; authoring the social primitives of *CiF* (relationships, social networks, statuses, the SFDB, and the CKB); the cast and initial story state; social norms of the storyworld; and social exchanges.

Following the general process outlined earlier in this chapter, a significant amount of authoring energy was devoted to determining the types of social behavior the students could engage in, how these behaviors would affect the student's relationships to each other, and determining the basic kinds of relationships we wanted our system to represent. Though a user of *CiF* could simply employ the power of their imagination to fill in the components of the *CiF* architecture to whatever their heart desires, the authors agreed that for the purposes of *Prom Week* there was no need to re-invent the social wheel, especially for a space that has been as richly explored as the high school social scene. Countless pre-existing media experiences —including novels, films, and plays— have already been authored that evoke some of the plights and pleasures of high school life; as opposed to fabricating our own behavioral rules, we leveraged these to construct a foundation for typical high school behavior. This served a dual purpose: in addition to informing the types of experiences we wanted our system to be able to model, it also shifted our focus away from other experiences that were not represented in the media we studied. Thus, they simultaneously provided direction and constraint, rendering a potentially intractable social space into something much more authorable. For the purposes of *Prom Week* we primarily relied on two pre-existing media sources: the film adaptation of Stephenie Meyer's *Twilight* of the same name (Hardwicke 2008) and *Mean Girls* (Waters 2004), a film adapted from Rosalind Wiseman's non-fiction *Queen Bees and Wannabes* (Wiseman 2003). Between these two films, we found a wealth of knowledge pertaining to social cliques and interpersonal relationships that coincided beautifully with the types of stories we wanted to tell.

Our work with the media largely focused on operationalizing them; analyzing the films as a whole, specifically the social interactions between characters, and reducing them into their component parts, such that the elements of social exchange could be encoded into algorithmic rules. In addition, a certain amount of exaggeration was woven into the dialogue to ensure that the message of any given exchange was made readily apparent to the audience. As the source material could already be considered exaggerations of "real" social conduct, the dialogue of *Prom Week* is somewhat removed from how a typical high schooler might actually speak. This is an intentional decision; the quirky nature of the dialogue's "voice" is intended to clarify changes in game state, as well as to entertain.

4.1 Analysis of Twilight and Mean Girls

The first design impetus we had when authoring for *Prom Week* began was to capture patterns of high school interaction with emphasis on dramatic content. To this end, we followed in the footsteps of the ethnographic-style analysis that was done on *Sex and the City* with sources more amenable to the types of dramatic interactions between teenagers we wanted the authoring to capture. Contemporary and popular pre-existing media artifacts we chose to be the basis for the analysis were indicative of our design goals; quickly changing, highly dramatic interaction between high school students with examples ranging from frivolous to serious were apparent.

As mentioned above, the two specific media artifacts that we made lengthy analyses of are the movies *Twilight* and *Mean Girls*. Each had strong examples of romance, dating, friendship, loyalty, enemies, vitriol, and easily observed patterns of normal social behavior. As these are fictional works that have been vetted as mass produced media artifacts and by popular culture, the distillation of dramatic situations with interactions that a witness can easily identify with was done by the authors of the works. For the purposes of gathering material on which to base our authoring, these pre-refined sources of high potency interactions between teenagers were excellent for quickly identifying what mattered in the social world to these types of characters. By applying ethnographic analysis on these media sources, we were able to draw out the salient meaning and knowledge and use it as the first step in building a world with the methods of representation afforded by *CiF*.

This analysis of *Mean Girls* and *Twilight* took the form of a "blue book party" at which a detailed analysis that spawned many points of clarification and discussion was held. The team of analyzers consisted of several graduate and undergraduate students equipped with note taking tools in a room with a television playing copies of the movies. At intervals ranging from scenes to dramatic beats, we paused the movie and took notes about what just took place. Emphasis was placed on looking at the scene through a dramaturgical metaphor. As we were looking to gather information about the social norms, character motivations, relationships, social status, and patterns of social interaction whose goal was to change the social landscape, each pause resulted in conversation and segments were frequently played back for additional information and clarity. Often the type of information gathered took the form of observations of behavior based on a social condition or character motivation. The following are a few examples of our observations:

- **Who is the Most Miserable:** When the teacher and principal talk about their summers, he got carpal tunnel syndrome, and she got a divorce, so she won the misery contest.
- **Assuming the Associability of traits:** When Lindsay saw the table of African American students, she got really excited because she thought that they would be able to speak another language (though they couldn't and gave her a cold stare)--assumed that people who looked the same/came from similar backgrounds would share similar abilities!
- **Insult Introduction:** Lindsay's alternative friend is first introduced to the audience and Lindsay through her being insulted (about her hair, I think), and her insulting the other person back.
- **Bad Influence:** One of the very first things that Lindsay's new friends have her do is ditch class and go hang out on the lawn. Even though Lindsay is a good student, she is so desperate for friends it appears that she decides to go for it.
- **Desires Unity:** This was a big driving force for Lindsay throughout the movie. Sometimes she lost sight of it, but by and large she wanted everyone to get along, and she wanted everything to be together. In my mind, this is connected to her statement about Math: she likes Math, because it is the same in every country.
- **Recruiter:** The idea that someone is willing and eager to try to bring other people into the social group.
- **Regulation Hottie:** An attractive person, with connotations of being associated with the 'hip' scene, the 'in crowd.' (female only? Traits based on gender?)
- **Skeeze:** Maybe this is the same thing as 'raging hormones' -- primary interested in getting into people's pants (in the movie it seemed to be male only?)

These types of social conditions and motivations for character action seemed to be intricately intertwined with long term social status, labels, and personality traits of the characters. As is the case with the *Not Like the Others* game from *Sex in the City*, different characters often employed similar social interaction patterns but used them in subtly different ways and with different a tact.

This type of information about what the characters cared about and what social situations prompted characters to act provided a conceptual foundation for understanding their interactions. With a general sense of what characters were motivated by, what they wanted to change in the world, and how they wanted to represent themselves, the patterns of social interaction came in to focus. If the goal was either to explicitly change the social state or to implicitly reinforce a social role, varying levels of character interactions found in the scenes and beats of the movies readily fit in to the abstraction of social exchange. Furthermore, labeling social exchanges with common interpretations (such as "mean", "nice", and "lame") seemed natural during the analysis. For example here is a "mean" labeled social exchange:

- **Unintentional Burn**: Two characters say something, which makes a third character feel bad, even though it was not the intention of the first two characters at all (Bella and girl A make a reference to eating disorders oblivious to girl B, but girl B happens to be eating a sandwich and looks crestfallen after hearing). Another instance was between only Bella and Edward -- referencing the weather Bella said that she hates cold wet things, but Edward--being a vampire--takes it personally (though Bella did not know he was a vampire at the time)

Because the design of *CiF* was being solidified in tandem with this analysis, common structures present in the analysis helped shape the representation and procedure used in *CiF*. Guided by dramaturgical analysis, life games, and motivational analysis, the basic *CiF* structures emerged. Particularly the primitives of relationships, statuses, personality traits, social networks, and social exchanges were prominent in this first analysis.

4.2 Social Primitives with *CiF*'s Abstractions

This section is a guided tour of authoring a world with *CiF*'s knowledge representation primitives. Periodically, the details of how *CiF* and *Prom Week* co-evolved and design insights gained from the development process are presented.

4.2.1 Relationships

A good first step when modeling a world of social interactions is determining what types of relationships the author wants the stories told by the system to be about. By relationship, the authors refer to a reciprocal state between two people that holds some amount of permanence; a large part of the play is to either forge new relationships between characters or disrupt old ones. Therefore, these relationships are a central focus of the gameplay and stories told. Picking good relationships is the key to creating a distinct storytelling space. Though *Prom Week* only uses three relationship types, more or fewer could be utilized to capture a desired social space. The three types of relationships found in *Prom Week* are:

Dating: The two characters are going steady. They are romantically interested in each other, and may even hold hands in public on occasion. If a character begins dating multiple characters at the same time, they will be marked as a "cheater," which would lessen the character's respect from the general public, though it may actually increase it amongst certain incorrigible youths.

Friends: The two characters get along well with each other. They are likely to like and respect each other, and if the romantic interest is there, this could be a first step on the road to the two characters dating each other.

Enemies: The characters do not get along well with each other. Social exchanges between enemies can be expected to be punctuated with insults and anger. Particularly vitriolic adversaries may even come to blows.

As mentioned above, all relationships are reciprocal. If character Lucas believes that he is friends with character Simon, then Simon is guaranteed to be friends with Lucas as well. (Non-reciprocated attitudes towards other characters are known as *character statuses*, and will be discussed below.) These relationships help motivate both character and player action—it is interesting to see how a tenuous friendship might devolve into fighting, or how two enemies may develop feelings for each other and ultimately begin to date. However, it was intended that each of these states represent some amount of permanence in the world, and achievement on the part of the player. Getting characters to change their relationships in some way is intended to feel like an accomplishment that is reached through investment in playtime and clever manipulation of the social space, and should not be something that fluctuates quickly with minimum effort. Even in our exaggerated context, it would be unrealistic for two bitter enemies to renounce their animosity and begin dating after a single social exchange, a second interaction to devolve them to merely friends, and a third to revert them back to enemies. Shifts in relationships should be the culmination of a rich history of interactions between two characters. Therefore, we need characters to be able to interact with each other and play meaningful social exchanges which build up this history and further the development of the characters without the interactions necessarily changing the relationships themselves. This need is in large part the inspiration behind the creation of social networks.

4.2.2 Social Networks

Though the term social network might conjure up images of Facebook and MySpace[43], social networks in the context of *CiF* are meant to model high level general attitudes the characters hold towards each other, such as how much respect or romantic interest a character feels towards another. Unlike relationships, the values of a social network are

[43] These social networking sites build social graphs around their users and are distinctly different from *CiF's* social networks.

very malleable, and are constantly shifting after every social exchange played. Thus, in addition to interactions which change the relationships of the characters and are meant to feel like milestones in the narrative of the system, there are interactions which only change the social networks, serving as stepping stones towards actually changing a relationship.

When authoring an experience using *CiF*, a good rule of thumb for selecting what social networks to use would be to determine the general motivators that might influence characters' desires to engage in the relationships decided upon. *Prom Week* incorporates three social networks:

Buddy: This network represents the general amiability one character feels towards another. A higher value is likely to lead to the characters becoming friends, while lower values could result in the characters becoming enemies.

Romance: This network is symbolic of how much one character is romantically attracted to another. Consequently, a higher value is indicative of a desire to begin dating.

Cool: The final network in *Prom Week* denotes how much respect a character has earned in the eyes of another. Since the concept of respect in a high school setting is wrapped up with several other notions, such as *street cred*, authenticity, and popularity, we determined that a network that represented how *cool* the characters find each other successfully captured these connotations.

Already we can see the system beginning to take shape. If the player has a scenario goal of getting two characters that currently lack interest in each other to start dating, potential courses of action suggest themselves to manipulate the social state. A straightforward idea to accomplish this would be to instigate social exchanges between the two potential love birds which raise their Romance Network values. However, alternative roundabout methods could be pursued as well. Perhaps if the user wishes for Lucas and Simon to date, she could have Lucas raise up his Buddy and Cool network values with all of Simon's friends. If all of his friends think highly of Lucas, then Simon could potentially be more interested in Lucas himself.

This speaks to another prime reason why we chose to ground our interactions in well-known pre-existing media experiences. By presenting users with recognizable situations based on media sources, players will be able to leverage their familiarity with the genre to conceive unique methods to manipulate the social landscape. In this way, even first time players can have an intuitive sense of exchanges which could result in Lucas and Simon dating. And just as either a direct or roundabout method can be used to successfully adjust the social networks for Simon and Lucas's romantic affair, so other kinds of social changes can be accomplished by multiple routes.

4.2.3 Statuses

Statuses can be thought of as temporary, non-reciprocal relationships. For example, let us say that Simon spurns Lucas's advances. Lucas, feeling rejected, may develop a sense of enmity towards Simon. Again this feeling is one sided—Simon may or may not feel any negative opinions towards Lucas—but Lucas's enmity towards Simon can affect all of his other interactions with him, if enmity is present in a social exchange's influence rule sets or preconditions. Statuses will dissipate on their own (as more entries are added to the SFDB and the event which sparked the status is relegated further and further into the past), and some social exchanges, such as paying a compliment, might patch up any ill feelings all together.

In addition to resulting from single social interactions (i.e., social exchanges), Statuses are also triggered from multiple entries in the SFDB. For example, a situation where multiple characters are picking on one character could result in the picked on character becoming depressed. While any one of these actions would not be enough to cause this status, the culmination of them, represented by a status trigger rule that will recognize when more than three different characters lower their buddy network values with a single character in a given time, create the status.

4.2.4 *Cultural Knowledgebase*

To help add character-specific flavor to these interactions, we make use of a CKB. The CKB allows for the dialogue that the characters share with each other to reference objects and concepts specific to the context of the world's setting (in the case of *Prom Week*, a high school). The CKB is populated by the author with objects appropriate to the setting that she wants to model. Each object is assigned a descriptive label such as cool, lame, romantic, etc. This label represents the zeitgeist; the *absolute truth* of the world. For example, the zeitgeist labels homework and cafeteria food as lame, and skateboards and cell-phones as cool. The second label category represents individual characters' thoughts on specific objects, which can—and often do—conflict with the vision of the zeitgeist. Characters can either like, dislike, or not have an opinion about an object. For example, if Simon is a bookish type, he might like homework, even though the zeitgeist regards it as lame. The system can then determine that Simon likes a lame thing, which can then be utilized in social interactions. If Lucas were to engage in an interaction with Simon with the intention of lowering the value of his "cool" social network with others (perhaps their romance has fizzled out), Lucas might specifically cite Simon's predilection for homework while insulting him as an example of why he is lame.

This not only helps provide a bit of variety with dialogue exchanges between characters, but it becomes an important way to learn more about the characters themselves. Through exchanges with other characters, the player could learn that both Lucas and Simon like the (generally regarded as lame) cafeteria food, which provides the player with a hint that the two of them have at least a little bit of foundation to get along. And indeed, the two could engage in an interaction in which they excitedly share their mutual interest, an interaction that could not be had—at least not successfully—with characters with completely dissimilar passions. As information is revealed about the characters, it adds both to the richness of the world, and to the player's ideas for how to go about manipulating it.

4.2.5 Social Facts Database

Another way that specificity can be inserted into social interactions is through the SFDB. This data structure captures information about every single social interaction that transpires throughout the course of the game. In doing this, the specifics of each social interaction, no matter how small, can help contribute to future exchanges between characters. This helps realize the goal of having distinct paths to accomplishing the same result (e.g., the direct path or a roundabout path to dating, as described above) having distinct impacts. Specific interactions can have labels associated with them which can then influence how characters engage with each other in the future. An insult from Lucas to Simon, for example, would be considered a negative act, and in future interactions between

Lucas and Simon, the SFDB will be looked at and, when it is discovered that Lucas had been mean to Simon, it will affect the outcome of their exchanges—and perhaps even lead to the specifics of the insult being brought up again.

CiF can also reason over multiple entries in the SFDB to discover particular social states which provide additional richness to characters' opinions of each other and knowledge of the world. Let us say that Lucas is dating Simon, but then Lucas successfully starts dating another person as well. By noting that there are two simultaneous dating entries for Lucas in the SFDB, the system will then mark Lucas as a cheater. This could have many varied repercussions down the line—it would certainly negatively impact his relationship with Simon, and it might prevent him from successfully dating other people in the future—word travels fast in this high school, and once it has been revealed that Lucas is a cheater, only those willing to give the two timing dog a chance at redemption will be willing to share their hearts with him.

The SFDB also serves as a means for the author to fill the world with backstory. The SFDB can be pre-populated, before the game begins, with specific social interactions which represent how the initial social state came to be. Entries could describe Lucas and Simon as friends who first met by sharing a mutual interest, but who have a caustic relationship marked with exchanging insults with each other. The player will at first only see a friendship relationship between the two characters (perhaps with a mutually low value on the buddy net to reflect the sting of the insults), but through play references to these entries in the SFDB will eventually reveal the characters' back-story. We believe this to be a relatively unique way of creating the background of a world—it is explicitly written in the language of in-game actions (a back-story SFDB entry will take the exact same form as an entry inserted during actual play), though the way that it is revealed in the discourse of the narrative is not explicitly up to the author, and depends entirely on how the player plays the game.

4.2.6 Characters and Traits

As described above, the individuality of characters in *CiF* is defined through their relationships to the objects in the world. Additionally, characters can be assigned traits that dramatically affect their motivations and reactions to the social world. As will be discussed below, traits are static characteristics that contribute to the ranking and results of the social interactions.

In addition to traits, *Prom Week* represents characters with animated avatars and a paragraph of prose to briefly describe the character's history and personality. However, these aspects of character representation depend on the interaction model of the experience that *CiF* is driving.

4.3 The Characters, Initial State, and Backstory

Building an initial social state for *Prom Week* required creating content for many of the knowledge representation areas of *CiF*: the cast of characters and their traits; the initial relationships, statuses, and social network values; a CKB; and a detailed and functional (in terms of being usable by *CiF* and containing interesting content that factors in to the character's reasoning during an evolving gameplay session) SFDB.

For cross-story consistency (so a player could "get to know" the characters of the world and their default relationships) and to provide a clean slate to author levels with, the initial social state of *Prom Week* was authored to be the starting state for every story. This gave authoring the initial state a tension between being understandable and socially stable (so that characters would not act autonomously or that few triggers would fire right after the first social exchange was performed) enough to provide a foundation for the player while having a good amount of existing dramatic tension.

The initial state required continuous small adjustments as stories were developed. For example, Simon's story had a story goal of having him become friends with Naomi to leverage her popularity to better his chances of becoming prom king. However, in the original configuration, it was difficult for players to make Naomi and Simon friends in the initial state as it was when this story was being developed. After some experimentation, Naomi's buddy network value toward Simon was increased from a low value to a moderate value and she was given the personality trait "Sympathetic" in addition to possessing the "Kind" trait. These changes made the difficulty of achieving Simon's story goal more in line with an earlier level.

A potential problem with tuning the initial state to balance levels is the risk of modifications having negative impact on the other stories. Fortunately, this was never a serious problem for *Prom Week*. I attribute this to the relatively large number of characters in the cast when compared to the number of stories. This created a social space large enough that minor tweaks for one story resulted in relatively minor impacts on other stories. There were only a small number of occasions when complimentary changes had to be made. Specifically, Nicholas was becoming the antagonist in a few of the stories and his relationships with other characters were becoming progressively worse (leaning more toward low social network values, less friends and dating relationships, and more enemies) in the initial state. This made Oswald's story more difficult as it was designed to be easier if he and Nicholas started dating; Nicholas had become significantly less desirable to date in the eyes of Oswald due to tweaks made to Nicholas. The situation was easily remedied by increasing all three of the social network values in both directions between the two characters and slightly increasing Nicholas' overall cool network over the entire cast. Now

Nicholas was still a *Prom Week* villain but that did not stop Oswald from finding Nicholas romantically desirable.

4.4 Microtheories

Authoring the microtheories took the form of asking "what if" and "what would you do" in abstract social situations as role playing exercise. This method worked well for us and we later learned that there was research done in using role playing as a tool in practicing ethnomethodology (Dennis 2009). We constructed the microtheories according to the authoring order set out in the Method for Authoring a Storyworld with *CiF* section: starting with relationships followed by social networks, statuses, traits, SFDB entries, and finally CKB entries.

The final result was the answer to this question: “how important is this situation to this social exchange intent?” Answers were in the form of a valence (positive or negative) and a magnitude, "not a factor" to "extremely important". To keep the weighting system consistent, the weights were typically in the range of -5 to 5; -5 was negatively “extremely important”, 0 was not a factor, and 5 was positively “extremely important.” When norms were critical to the performance of a social context, they were given weights with a magnitude of up to 15. This extreme weighting relative to the [-5,5] range all but ensures the result of desire formation or the responder’s choice will reflect the weighting. However, a completely overwhelming weight or a forcing of desire was intentionally not used as we wanted a large number of smaller factors to have the possibility of outweighing a few large ones.

A view of the microtheories in their final state can be seen in Figure 46. The relationship microtheories have the most influence rules and contain a lot of variety in the weights of those influence rules. This is to be expected as relationships are important to almost all reasoning done in *Prom Week*. Microtheories over social networks had a small number of rules but a similar variety of weights. Statuses have a smaller amount of more focused, highly weighted rules. These microtheories are not included as often as other types but have significant impacts when they are used. A good example is the microtheory for the Cheated On By status. It is almost entirely composed of influences rules with weights of -5 or lower. The "someone did something mean to me in the past" microtheory has strongly negative weights to friendly intents and strongly positive weights to the intent to enemies. Microtheories with compound definitions were often created to add more detailed reasoning to certain social situations. During player testing, we found the characters needed more social knowledge about dealing with someone who was cheating on them. To this end we made a microtheory about dealing with someone you are dating and that person is cheating on you; it contains a lot of highly weighted negative influence rules.

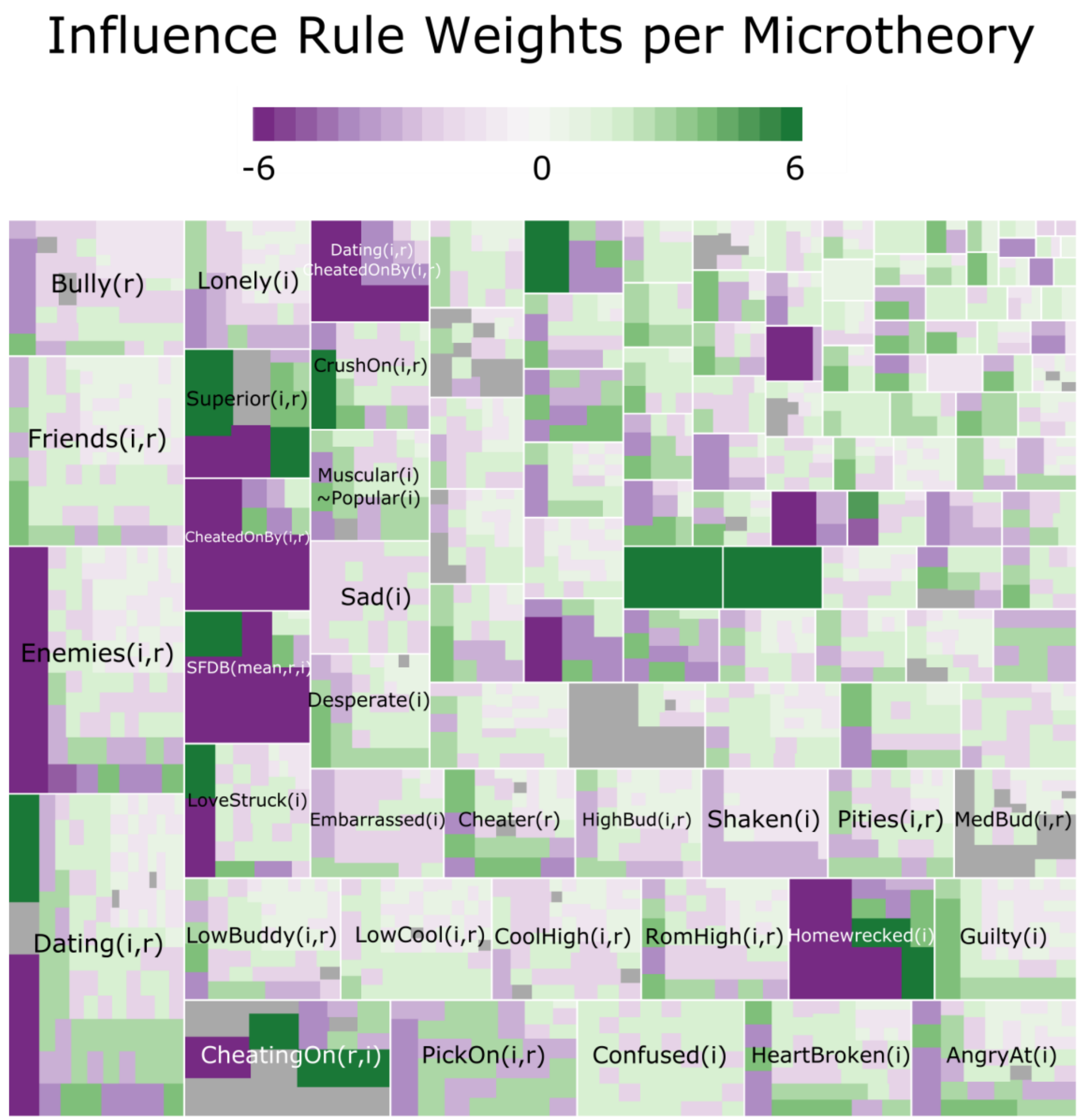


Figure 46 – A treemap of influence rule weights grouped by microtheories. The boxes separated by white borders are the boundaries between microtheories (labeled by their definition). Sections of each microtheory rectangle are colored and sized according to the weight of the influence rules. Grey areas are to be ignored as they are artifacts of the treemapping process and represent empty space. Microtheory labels that were too large to fit were removed.

4.5 Social Exchanges and Instantiations

The principle that guided the authoring of social exchanges was to achieve broad coverage while providing depth as required. Broad coverage refers to having social exchanges (specifically their abstract form and not their instantiations) cover all ways a character can change the social space (e.g. the 12 intent types) and have those exchanges match the expectations of the player (provided by the microtheories). Depth of social exchanges refers to the specific social contexts that instantiations and their social change and conditions represent. We found that some social situations required a large number of instantiations which are reactive to a fine level of social detail. An example is when a character is dating multiple people and gains a cheating status, players expected a majority of subsequent interactions to address the fact that cheating happened. If the character performed otherwise, they were seen as either breaching the player's sense of social norms or lacking in social knowledge. Other situations, such as becoming friends, required less attention to the details of social context in performances; social exchanges subsequent to making characters friends were not expected to address the event.

Each instantiation within a social exchange is unique and can change the social world in unique ways. Communicating this change is important to helping players build mental models of the social world and is done implicitly (through dialogue and animations) and explicitly. The first way change is communicated explicitly is through the social exchange results UI. After a single exchange is played, changing the social world and revealing what it does to the characters, the world subsides. This gives players an opportunity to reflect on the story so far, examine the next possible actions, and decide where to take *Prom Week*'s world next. The second way social change is explicitly communicated to the player is by building the final social change from partial changes during performance. This consists of social change associated with lines of dialogue. The zing of a pickup line failing would be accompanied by an iconic indication of the romance network down or an embarrassed status being gained.

The partial change rules provide an additional affordance to the author as they are not restricted to the final change of the instantiation; the only restriction is that an instantiation's partial changes should add up to the final change by the end of the performance. This use of partial change will be illustrated in a continuation of the pickup line example. In this instantiation of the Pick-Up Line social exchange, the end result is the lowering of the responder's romance network value toward the initiator. To add more dramatic content, the initiator could have chosen a really bad pickup line and the partial changes of lowering the responder's romance network to the initiator and the initiator gaining the embarrassed status are shown. In the next line of dialogue, the initiator apologizes for saying something that uncouth. This is accompanied by a partial change of removing the embarrassed status. The instantiation then ends with only a lowering of the responder's romance network value toward the initiator; the status of embarrassed was only used for dramatic effect.

Depth of social exchanges is also impacted by the way social exchange preconditions and instantiation conditions are used to choose appropriate social contexts. If used overzealously, they can cull too many possibilities and will rarely be seen. Thus, conditions are only to be used to prevent scenarios, and an author should be careful not to limit interesting dramatic scenarios by encoding too many precondition or instantiation conditions.

Instantiation conditions are also used to set the state for performances. If an SFDB or CKB entry is referenced in an instantiation's NLG template, it must also be present in the instantiation's condition. Otherwise the author runs the risk of trying to reference something that does not exist in the storyworld. As this authoring mistake happened periodically in the development of *Prom Week,* we mitigated this by having a sensible performance default for when SFDB and CKB queries in NLG templates fail: "*whisper whisper*" for SFDB and "thing" for CKB failures.

4.6 Story Endings

Though gameplay use of story endings was covered in chapter 4's Story Endings section, creating content for their actual performances was an authoring challenge distinct from creating performances for social exchanges. The difference lies in the constraints of two venues: social exchanges instantiations are meant for use generally (by any initiator) and allow for the story to be continued, while story ending instantiations typically have a set initiator (the lead story character) and are meant to be a dramatic finale to the story.

Also mentioned in chapter 4 , story endings can end one, many or no story goals. The rest of this section is dedicated to various examples from Doug's story that shows story endings for a variety of story goal conditions (Table 14 in chapter 4 shows the complete mapping of story goals to story endings for Doug's story). The first example is the Kickin It story ending which is performed when no story goals are met:

Responder: Oh, hey Doug! %greeting%! I'm a little surprised to see you here!
Doug: Why would you say that?
Responder: It just doesn't seem like the Prom is really your type of thing, yah know?
Doug: Oh, yeah. Totally. I just kinda had a bit of a change of heart yesterday - I guess.
Doug: I mean, like, I may not have achieved everything I wanted to. But that's just me, man. I'd rather chillax than stress out over deadlines.
Responder: Oh, Doug. I'm so going to miss you when I'm famous.

This example is named Little Kindness, Long way and covers the Nice Guy story goal and refers to something positive that Doug did in the past in the topic of conversation tag:

Responder: Yo Doug... uh... so I was thinking about that time when %toc1%.

Doug: Yeah?

Responder: That was just... really cool of you, man. Sometimes... I wish I could be that guy, who does nice things.

Doug: There's nothing stopping you, man. Nothing at all. Just be yourself.

Responder: Uhh.. heh heh... but what if deep down, you know, I'm really just a total jerk who likes making fun of dorks? No offense.

Doug: Sorry, man, you have to figure that out for yourself.

Responder: Like algebra.

The Burning Bridges ending refers to something negative Doug did in the past and ends the Kinda Mean Dude story goal:

Responder: Doug... some of us have been kinda worried about you. Is everything all right?

Doug: Yeah. Why wouldn't it be?

Responder: Well, like that time when %toc1%? That was pretty un-Doug.

Doug: What's it to you? Who are you to tell me what I should be like? Why don't you leave me alone, dipweed!

Responder: Doug... We're just worri-

Doug: Well don't be. Just leave me alone, alright? Is that 'Doug' enough for you?

Responder: I... I'm sorry. I guess it's easier to just put each other in little boxes than try to actually understand what someone else is going through.

Doug: Or maybe I'm just more of a bad boy than you losers ever gave me credit for. Hey, enjoy your Prom. I'm out.

When a player achieves multiple goals in a story, they are rewarded with story endings that are more dramatic and a little longer. Bad Boys Get All the Babes is a story ending that refers to something negative Doug did in the past and ends both the Kinda Mean Dude and Last Minute Date story goals:

Responder: Well, if it isn't my sexy, little stud-muffin Doug.

Doug: Hey babe. Yeah, thought I'd drop in, see what the big deal is about this prom thing. Not that I care.

Responder: Why don't you spend it with me, love? I'd just die to dance with you tonight.

Doug: Eh, I dunno. I was maybe gonna skate downtown, break some windows, see what kind of trouble I could get into.

Responder: Awww, don't go. Be a good boy and stay here with me. It's our special night!

Doug: Special? Whatever. It's just a meaningless ritual to distract us all from the fact that now we're unemployed and have learned no useful skills. Let's go burn something.

Responder: You're so hot when you get all bad, stud muffin. Hey, remember that time when %toc1%?

Doug: Someone had to do it, baby. Someone had to do it.

The last example story ending, Something Special, features a "feel good" instantiation and addresses the Big Buddy Doug and Last minute Date story goals:

Responder: Man, I can't believe we're finally here at our senior prom. We made it!

Doug: Totally, %gender(r,my handsome man,lady love)%. And I couldn't imagine spending it with a more awesome %gender(r,dude,babe)%.

Responder: Same here. When this week started, I never thought I'd end it with such a sweet, kind, handsome man.

Doug: Aww... you're makin' me blush.

Responder: I feel like we have something that's gonna last, you know? I mean, this isn't just a fling. We're friends too.

Doug: Next you're gonna say we should get %gender(r,twinner,his and hers)% skateboards.

Responder: You're on, hot shot.

Doug: Hey... can I kiss you?

Responder: You damn well better.

Chapter 8

Conclusion and Future Work

1 Summary of Contributions

This dissertation has presented research on the design, development, and usage of the AI implementation of a playable model of social interaction, *CiF*. This system also represents an authoring approach, which has been conceptually situated in social science, whose use is described in this dissertation. The design of *Prom Week,* the experimental game whose design, mechanics, and player affordances are tightly coupled with the capabilities of *CiF*, has been described. *Prom Week* has been evaluated with respect to this dissertation's stated research contributions. The real-time and non-verbal aspects of a playable model of social interaction along with the concept of supporting character realism have been discussed, developed, and evaluated in the *Holodeck* chapter. Finally, this dissertation described a method for authoring content with *CiF* complete with case examples and design insights.

2 Future Work

The research presented in this dissertation opens up many possibilities for future work. In keeping with this research agenda, there are refinements that can be made with respect to the research contributions, additional fictional and virtual worlds that can be created, and user studies to be performed. This work can be expanded to explore new playable models and to form a methodology to explore a game design process heavily integrated with AI system design.

2.1 Integrate CiF and CiF-RPG

A short term project (that is already underway) is to merge *CiF* and *CiF-RPG* into a canonical version of the system. As both branches have grown independently, their integration will entail a detailed merging. By merging the two, the new version of *CiF* will have all of the existing capabilities (as described in chapter 3) as well as the capabilities developed for *CiF-RPG*. As the name suggests, *CiF-RPG* was used in the creation of the non-combat CRPG *Mismanor*. *CiF-RPG*'s includes functionality for items, a notion of game objects that includes both characters and world objects, a sense of character location, and character inventories.

2.2 Prom Week User Studies

The evaluation presented in chapter 5 is only a first look at assessing *Prom Week*. Because of the new space of play and new level of player choice complexity enabled by *CiF*, a detailed look into how players interact with artifacts in this new space of design would provide additional insight and ways to improve interactive experiences made with *CiF*.

On possible user study would be to measure narrative involvement, distraction, ease of cognitive access, narrative realism, empathy, sympathy, cognitive perspective taking, loss of time/self-awareness, and narrative presence through a survey (Busselle and Bilandzic 2009). A similar questionnaire could be given over the players' identification with the characters (Vorderer and Bryant 2006). As there are results of this same survey for other media experiences, one possible point of analysis and evaluation would be to compare the survey results of *Prom Week* to those other experiences. This could be reinforced with a few deeper qualitative interviews to get a deeper understanding of how *CiF* and *Prom Week* convey fictional worlds.

2.3 AI-Based Game Design

AI-based games are games in which the AI systems and the core game mechanics are profoundly interconnected, often to the extent that they are indistinguishable from each other. *Prom Week* is a clear example of this type of design as its gameplay affordances and game mechanics are taken directly from the capabilities of *CiF*.

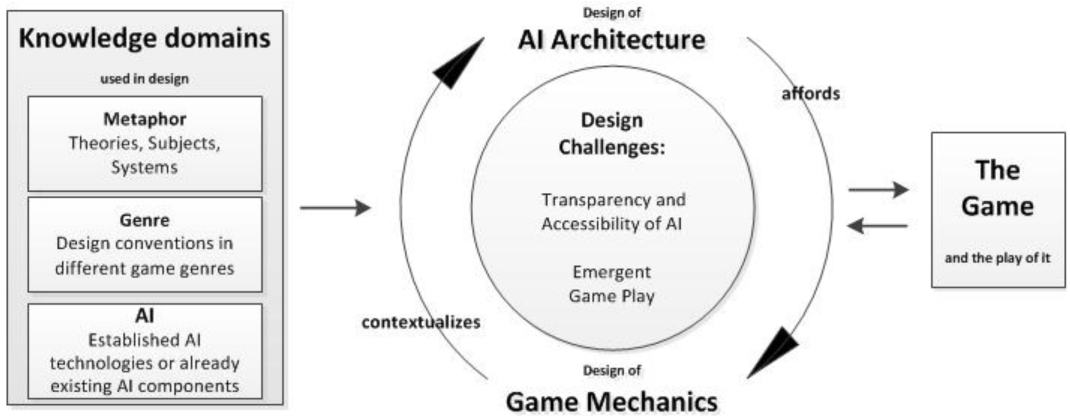
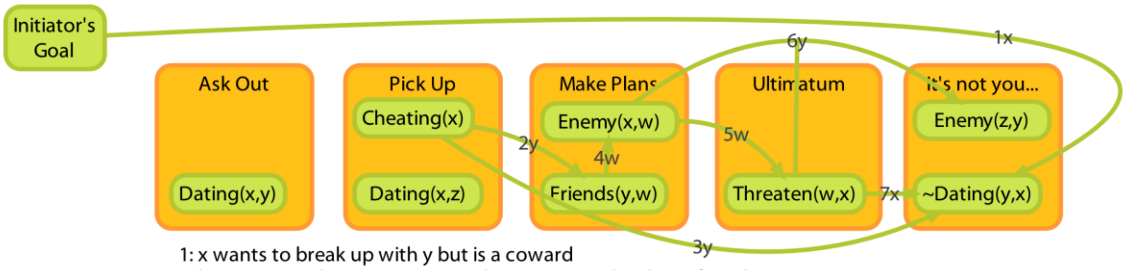


Figure 47 - An overview of the process of AI-based game design.

Game design is a process that most often is conducted in iterations (Salen and Zimmerman 2001) where each iteration informs the design of the next. When AI components are part of the core game play, the design of the components needs to be incorporated in the iterative process. Figure 47 illustrates the process of AI-based game design. A key stage in this process is to understand the systems' design affordances, i.e. the actions supported by the nature of the system design. The affordances given by the AI components and the game design shape the final outcome of the design process: the game.

2.4 Abstractions of Social Interaction

Though a useful representation space for building social puzzles and exploring the hallway politics of high school, there are levels of coherence and social interaction are not



1: x wants to break up with y but is a coward
2: because x is cheating on y, y seeks support and makes a friend
3: because x is cheating on y, y wants to stop dating x
4: now that x and w are friends, x's cheating makes x and w enemies.
5: because x and w or enemies, w threatens x
6: because y's friend w threatened x and because z is dating x, z becomes enemies with y
7: w's threat to x and x's cowardice causes x to break up with y

Figure 48 – A sequence of social exchanges with the surrounding context of chaining dependencies.

represented by *CiF*. *Prom Week* has a static notion of longer term social world transformation via story goals. However, whole new types of game experiences could be made possible if the equivalent to the abstraction of social exchanges were developed and modeled for longer term interactions. If a generalized method for using social exchanges as the unit of social change (much like predicates are the unit of change in social exchanges) was developed, a new level of procedural narrative could be reached.

The limited duration, limited participation social exchanges currently in *CiF* are like small islands floating in a larger sea of social context. They are loosely situated in social plausibility by preconditions, microtheories, and sets of influence rules but are only vaguely connected to one another. To increase the power of social exchanges, the connections to and patterns among these islands need to be discovered and represented.

One way to view the context around these explored islands is to look at the reasons why players choose strategies to pursue to get to the goal of scenarios in *Prom Week*. Figure 48 shows an example of a scenario with the goal of a character, Edward, to break up with Phoebe. However, Edward has the cowardly trait and is unlikely to directly end the relationship. The player decides to have Edward play Ask Out with Mave which succeeds and results in them dating and Edward gaining the status of cheater. The player reasons that since Phoebe has been cheated on, she needs a friend for support so she plays "*let's hang out*" with Zack. A trigger rule picks up on the combination of Phoebe being cheated on by Edward and Phoebe being Zack's friend and results in Zack and Edward becoming enemies. The player decides to have Zack stick up for his new friend by playing the hierarchical social exchange "Mess Up and You Get Jacked Up" with Edward and gives Edward the statuses of Afraid of Zack and Afraid of Phoebe. To diffuse the situation, the player has Edward play Let's Be Friends with Phoebe. This social exchange accomplishes the scenario's goal.

This player-driven context could be used as a foundation to create a higher level of abstraction. The *CiF* authoring tool used to create the content for *Prom Week* could be adapted to have a hierarchical social exchange editing interface in which patterns of social exchanges could be entered and would look much like Figure 8. An alternative to this knowledge authoring approach would be to construct hierarchical social exchanges based

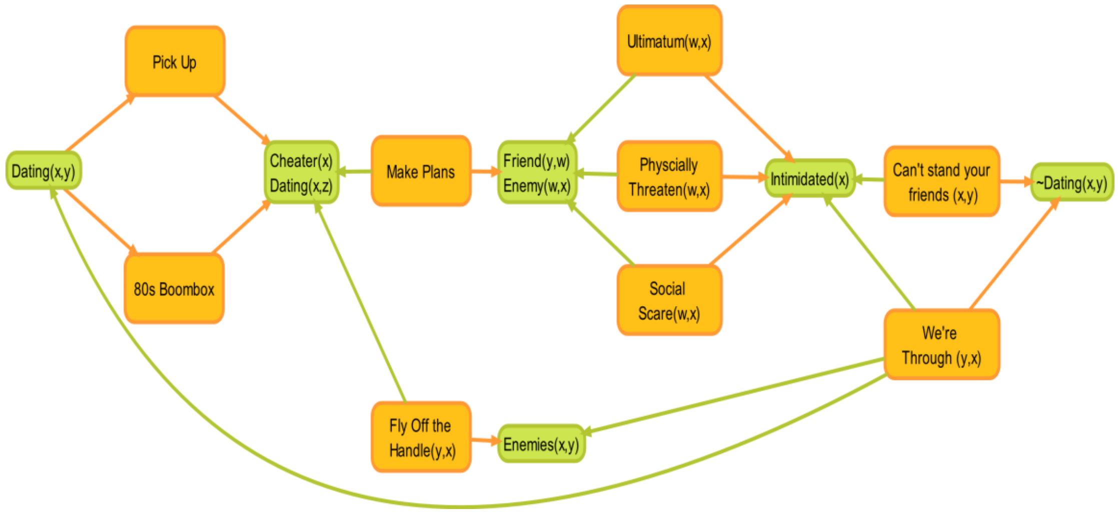


Figure 49 - An example of a small hierarchical social exchange. The orange nodes are social exchanges, the green nodes are the social state the games depend on, orange edges indicate temporal ordering, and green edges are dependency links.

on a corpus of game play. This approach of creating patterns of higher abstraction from data of a lower abstraction is currently being explored in a few research projects. Gameplay traces of *The Restaurant Game* are used by a semi-automatic methodology for recognizing tasks in those traces (Orkin et al. 2010) augmented by an annotation tool for non-experts. Because this approach has been proven to work in one game using a large number of traces, it could be adapted to a similar data set from interactive narratives (Gerber, Gordon, and Sagae 2010; Swanson and Gordon 2010). These data-driven approaches for creating higher levels of social exchange abstraction are desirable as there should be a large corpus of gameplay generated by the release of *Prom Week*. Since *Prom Week* includes both sandbox and scenario play modes, we will have gameplay traces that cover both open-ended and goal-directed patterns of social exchange play.

Similar to how social exchanges were viewed as a graph of events in early versions of *CiF* (McCoy and Mateas 2009), social exchanges could be "hierarchically" composed of other social exchanges. Instead of the nodes of the graph being discrete, atomic changes in social space, each node could be a social exchange. The edges would have to become more complex; in addition to showing directionality of the game, they would have to be more meaningful like causal links (Enberthy and Weld 1991; Riedl and Young 2004) and conditional branching. A hierarchical social exchange would have to reason over more than just social exchanges; details such as the social fallout, failure cases, and topics of discourse need to be taken into account (if two characters were bonding over an event in the past, that event needs to be kept constant through the hierarchical game). A prototypical example of a dependency graph of a hierarchical social exchange can be seen in Figure 49.

Through building and finding solutions to scenarios in *Prom Week*, strategies for building social puzzles have been developed. Though not identical, hierarchical social exchanges share aspects with scenarios. They have multiple paths to optionally multiple end states, social states (barriers) that need to be reached before progression is possible, and can tell a variety of narratives in a similar space. One way to tackle this complexity is to author backwards by starting at the end state and working to start states. This method promotes the discovery of a larger number of valid initial conditions. Considering the problem from a role-centric perspective helps authors keep the characters consistent when traversing the space. Often renaming the roles from initiator, responder and other prove helpful; cheating bastard, innocent victim, or best friend are often provide a better basis for creativity. For authoring a set of scenarios, a factoring of the knowledge gained from the ethnographic studies of the pre-existing media experiences was very valuable. The categorization and combination of scenarios' larger patterns is likely to help authoring hierarchical social exchanges just like it did when authoring social exchanges in *Prom Week*.

2.5 Integrate CiF with Other AI Systems

One motivation for integrating *CiF* with another AI system is to bring a playable social model into a real-time environment. A new level of character performance realization could be reached if the real-time non-verbal behaviors of the SCR avatars of the *Holodeck* could be tightly coupled with more abstract social reasoning. To this end, *ABL* and *CiF* would pair well.

As *CiF* has already been used to author and simulate fictional storyworlds, it could be used in conjunction with another system, such as *Story Canvas* (Skorupski 2009; Skorupski and Mateas 2010), to create a mixed initiative authoring tool. Player interaction in *Prom Week* is, in some sense, similar to authoring a story with *CiF* as an advisor. The AI contribution could be reduced to a level of simply complying with the human author's commands or to the other extreme of authoring the entire story with the human author having no input. The relative initiative of the human to the AI author is continuous space which has been explored by other systems. The goal of combining novice-friendly story authoring and social simulation is to enable a new space of stories that are in keeping with social norms. With the social complexity partially accounted for in *CiF* and the technical complexity reduced by *Story Canvas*, more complex stories could be created by an experienced author and the barrier to create quality stories by novices would be lowered.

3 Conclusion

In video game design, innovation in gameplay has taken a secondary role to graphical improvements, physical simulation, and aesthetic experimentation. Although gameplay innovation happens in places like the independent games community, there are regions of design space only explorable via this kind of work. One way to address this is by making new spaces of video games as playable as the current playable models of physics or combat. AI modeling techniques can be used to make more spaces playable and, thus, produce needed gameplay innovation.

CiF and *Prom Week* represent unique and novel innovations by making social interactions playable. As such, it enables previously intractable types of games to be created and new levels of player choice complexity to be experienced. The benefits of *CiF* include a new style of authoring social norms and common patterns of social interaction for fictional and virtual worlds. The development and release of *Prom Week* has resulted in both acceptance of the game's innovative qualities and a proof that *CiF* can be indeed be used in the construction of new types of playable experiences.

Outside of game design, this work exists as a fresh and unique interpretation of social science. It consists of more than simply borrowing ideas from another field; the creation of interactive models of human behavior founded in social science is a critical reading and reflection of the implemented theories. I believe such interdisciplinary work is of great benefit. The confluence of disparate fields of study has the potential to yield great leaps in innovation that would further all the disciplines involved.

It is my hope that this work helps shape shape the future of game design by promoting new types of playable experiences. Technical and conceptual innovation aimed directly at promoting new types of gameplay can only have positive results for both players and designers.

References

Agre, Philip E. 1997. *Computation and Human Experience (Learning in Doing: Social, Cognitive and Computational Perspectives)*. Cambridge University Press, July.

Alexander, Leigh. 2012. GDC 2012: New ideas in socially-engaging AI design. *Gamasutra*. http://www.gamasutra.com/view/news/164664/GDC_2012_New_ideas_in_sociallyengaging_AI_design.php.

Aylett, R. S., S. Louchart, J. Dias, A. Paiva, and M. Vala. 2005. Fearnot!: an experiment in emergent narrative. In *Proceedings of Intelligent Virtual Agents (IVA05)*, 305. http://portal.acm.org/citation.cfm?id=1099302.

Aylett, Ruth, Joao Dias, and Ana Paiva. 2006. An affectively-driven planner for synthetic characters. In *Proceedings of ICAPS*, 2006:2-10. AAAI Press. http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:An+affectively+driven+planner+for+synthetic+characters#0.

Aylett, Ruth, Natalie Vannini, Elisabeth Andre, Ana Paiva, Lynne Hall, and Sibylle Enz. 2009. But that was in another country : agents and intercultural empathy. In *Proceedings of AAMAS 2009*, 329-336.

Bailenson, Jeremy N., Andrew C. Beall, Jack Loomis, Jim Blascovich, and Matthew Turk. 2004. "Transformed Social Interaction: Decoupling Representation from Behavior and Form in Collaborative Virtual Environments." *Presence: Teleoperators and Virtual Environments* 13 (4) (August 13): 428-441. doi:10.1162/1054746041944803. http://www.mitpressjournals.org/doi/abs/10.1162/1054746041944803.

Bakker, J I. 2011. "The 'Semiotic Self': From Peirce and Mead to Wiley and Singer." *The American Sociologist* 42 (2-3): 187-206-206. doi:10.1007/s12108-011-9140-3.

Bales, Robert F. 1951. *Interaction Process Analysis*. *Technology*. Addison-Wesley.

Bell, David. 2009. "Learning from Second Life." *British Journal of Educational Technology* 40 (3): 515-525. doi:10.1111/j.1467-8535.2009.00943.x. http://doi.wiley.com/10.1111/j.1467-8535.2009.00943.x.

Benmergui, Daniel. 2012. Storyteller. Independent. http://www.ludomancy.com/storyteller.php.

Berne, Eric. 1964. *Games People Play: The Basic Handbook of Transactional Analysis*. Seventh Im. New York: Ballantine Books. http://www.amazon.com/exec/obidos/ASIN/0345410033/qid=1099166101/sr=2-1/ref=pd_ka_b_2_1.

BioWare. 2007. Mass Effect. Microsoft Game Studios. http://masseffect.bioware.com.

———. 2011. Star Wars: The Old Republic. Austin, TX: Electronic Arts. http://www.swtor.com.

———. 2012. Mass Effect 3. Electronic Arts. http://masseffect.bioware.com/.

Bioware. 2009. Dragon Age: Origins. http://dragonage.bioware.com/.

———. 2010. Mass Effect 2. http://masseffect.bioware.com/.

Blascovich, J, J Loomis, Andrew C Beall, Kimberly R Swinth, Crystal L Hoyt, and J N Bailenson. 2002. "Immersive Virtual Environment Technology as a Methodological Tool for Social Psychology." *Psychological Inquiry* 13 (2): 103-124. doi:10.1207/S15327965PLI1302_01. http://www.informaworld.com/openurl?genre=article&doi=10.1207/S15327965PLI1302_01&magic=crossref.

Blow, Jonathan. 2006. There's not enough innovation in games! In *Game Developers Conference*. San Francisco. http://number-none.com/blow/slides/rant_2006.html.

Blumer, Herbert. 1986. *Symbolic Interactionism: Perspective and Method*. University of California Press. http://books.google.com/books?hl=en&lr=&id=HVuognZFofoC&pgis=1.

Boal, Augusto. 1985. *Theatre of the Oppressed*. *Unesco Courie*. Vol. 50. Theatre Communications Group. http://www.theatreoftheoppressed.org/en/index.php?nodeID=2&country_id=42&continent_id=7.

Boal, Augusto, and Marie-claire Picher. 2007. "Democratic Process and the Theater of the Oppressed." *Practice*: 79-89. doi:10.1002/ace.

Bourdieu, Pierre. 1990. *The Logic of Practice*. Ed. Richard Nice. *Man*. Stanford University Press. doi:10.2307/2804264. http://www.jstor.org/stable/2804264?origin=crossref.

Bratman, Michael E. 1999. *Intention, Plans, and Practical Reason*. *CSLI Publications ISBN 1575861925 httpcslipublicationsstanfordedusite1575861925shtml*. Cambridge University Press. http://www.amazon.co.uk/exec/obidos/ASIN/1575861925/citeulike00-21.

Brookey, Robert Alan, and Kristopher Cannon. 2009. "Sex Lives in Second Life." *Critical Studies in Media Communication* 26 (2): 145-164. doi:10.1080/15295030902860260. http://www.informaworld.com/openurl?genre=article&doi=10.1080/15295030902860260&magic=crossref.

Bungie. 2008. *Halo 3*. [Xbox 360], Microsoft, September.

Buono, Paolo, T Cortese, F Lionetti, M Minoia, and Adalberto Lafcadio Simeone. 2008. A Simulation of a Fire Accident in Second Life. In *Proceedings of the 11th Conference on Presence*, ed. Anna Spagnolli and Luciano Gamberini, 183-190. CLEUP, Padova, Italy.

Bura, Stéphane. 2012. Storybricks in Beyond Eliza: Constructing Socially Engaging AI. In *AI Summit of the 2012 Game Developers Conference*. San Francisco.

Busselle, Rick, and Helena Bilandzic. 2009. "Measuring Narrative Engagement." *Media Psychology* 12 (4) (October): 321-347. doi:10.1080/15213260903287259. http://www.informaworld.com/openurl?genre=article&doi=10.1080/15213260903287259&magic=crossref||D404A21C5BB053405B1A640AFFD44AE3.

Cavazza, M, F Charles, and S J Mead. 2002. *Character-based interactive storytelling*. *IEEE Intelligent Systems*. Vol. 17. IEEE Computer Society. doi:10.1109/MIS.2002.1024747. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1024747.

Charon, Joel M. 2009. *Symbolic Interactionism: An Introduction, An Interpretation, An Integration (10th Edition)*. Prentice Hall. http://www.amazon.com/Symbolic-Interactionism-Introduction-Interpretation-Integration/dp/0136051936.

Church, Doug. 1999. "Formal Abstract Design Tools." *Game Developer* 3 (8): 28. http://www1.cs.columbia.edu/~cs4995/files/Doug_Church_FADT.pdf.

Coleridge, Samuel Taylor. 2006. "Biographia Literaria (1817)." *Search*: 4.

Colmerauer, Alain, and Philippe Roussel. 1996. The birth of Prolog. In *Computer*, 28:37-52. ACM. doi:10.1145/154766.155362. http://portal.acm.org/citation.cfm?id=1057820.

Corrington, Robert S. 1993. *An Introduction to C. S. Peirce*. Rowman & Littlefield Publishers. http://www.amazon.com/An-Introduction-C-S-Peirce/dp/084767813X.

Crawford, Chris. 2002. *The Art of Interactive Design: A Euphonious and Illuminating Guide to Building Successful Software*. No Starch Press. http://books.google.com/books?id=lefh5rSLbVUC&pgis=1.

———. 2004. *Chris Crawford on Interactive Storytelling*. New Riders Games. http://www.amazon.com/Chris-Crawford-Interactive-Storytelling/dp/0321278909.

Cushing, Frank H. 1998. *My Adventures in Zuni*. Filter Press. http://www.amazon.com/My-Adventures-Zuni-Frank-Cushing/dp/0865410453.

Dennet, D. 1987. *The Intentional Stance*. MIT Press.

Dennis, Barbara. 2009. "Acting Up: Theater of the Oppressed as Critical Ethnography." *International Journal of Qualitative Methods* 8 (2): 65-96. http://ejournals.library.ualberta.ca/index.php/IJQM/article/viewArticle/3819.

Dias, João, and Ana Paiva. 2005. Feeling and Reasoning: A Computational Model for Emotional Characters. In *EPIA*, ed. C Bento, A Cardoso, and G Dias, 3808:127-140. Springer.

Dörner, D. 2003. The mathematics of emotions. In *Proceedings of the Fifth International Conference on Cognitive Modeling*, ed. D.D. Frank Detje and H. Schuab, 75-79. Bamberg, Germany.

Eladhari, Mirjam Palosaari. 2009. Characterising Action Potential in Virtual Game Worlds applied with the Mind Module.

Eladhari, Mirjam Palosaari, and M. Sellers. 2008. Good Moods – Outlook, Affect and Mood in Dynemotion and the Mind Module. In *Futureplay*. http://sites.google.com/site/mirjame/publications.

Ellis, Judi. 1996. Prospective memory or the realization of delayed intentions: A conceptual framework for research. In *Prospective Memory Theory and Applications*, ed. Maria Brandimonte, Gilles O Einstein, and Mark A McDaniel, 14:1-22. Erlbaum. http://psycnet.apa.org/psycinfo/2002-02930-001.

Enberthy, J Scott, and Daniel S Weld. 1991. UCPOP : A Sound , Complete , Partial Order Planner for ADL. In *In Proceedings of the Third International Conference on Knowledge Representation and Reasoning*.

Erol, Kutluhan, James Hendler, and Dana S. Nau. 1995. Semantics for Hierarchical Task-Network Planning. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.26.6180.

Evans, Richard. 2008. "Re-Expressing Normative Pragmatism in the Medium of Computation." *Proceedings of Collective Intentionality VI*.

———. 2009. "The Logical Form of Status-Function Declarations." *Etica & Politica* 11 (1): 203–259. http://www2.units.it/etica/2009_1/EVANS.pdf.

———. 2011. "Using Exclusion Logic to Model Social Practices." *Agents for Games and Simulations II*: 163–178. http://www.springerlink.com/index/W2675776V4411H58.pdf.

Figueiredo, Rui, Antonio Brisson, Ruth Aylett, and Ana Paiva. 2008. "Emergent Stories Facilitated." *Interactive Storytelling* 5334: 218 - 229. doi:10.1007/978-3-540-89454-4. http://www.springerlink.com/index/x066lu2010676736.pdf.

French, J. R. P., and B. Raven. 1968. *Group Dynamics: Research and Theory*. New York, New York, USA: Harper & Row. http://www.amazon.com/Group-Dynamics-Research-Dorwin-Cartwright/dp/B001E6ZWO8.

Friedman, Doron, Anthony Steed, and Mel Slater. 2007. "Spatial Social Behavior in Second Life." Ed. Catherine Pelachaud. *Intelligent virtual agents* 4722: 252-263. doi:10.1007/978-3-540-74997-4_23. http://www.springerlink.com/index/g4175v2h4m3t0617.pdf.

Garfinkel, Harold. 1967. *Studies in Ethnomethodology*. Ed. Prentice Hall. *American Sociological Review*. Vol. 33. Prentice-Hall. doi:10.2307/2092244. http://www.jstor.org/stable/2092244?origin=crossref.

Garfinkel, Harold, and Anne Rawls. 2002a. *Ethnomethodology's Program: Working out Durkheim's Aphorism*. Ed. Anne Warfield Rawls. *American Journal of Sociology*. Vol. 59. Rowman & Littlefield Publishers Inc. http://books.google.com/books?hl=en&lr=&id=zlWVh15JNkkC&oi=fnd&pg=PR9&dq=Ethnomethodology's+Program&ots=hYqedw7SKL&sig=1ibdY5VUmTM-t4J7eYB5QvjiL_4.

Garfinkel, Harold, and Anne Warfield Rawls. 2002b. *Ethnomethodology's Program: Working Out Durkheim's Aphorism (Google eBook)*. Rowman & Littlefield.

Gerber, Matt, Andrew Gordon, and Kenji Sagae. 2010. Open-domain Commonsense Reasoning Using Discourse Relations from a Corpus of Weblog Stories. In *Proceedings of the 1st International Workshop on Formalisms and Methodology for Learning by Reading (FAM-LbR) NAACL 2010 Workshop*.

Goffman, Erving. 1959. *The Presentation of Self in Everyday Life*. Anchor.

———. 1974a. *Frame Analysis*. Ed. Keith Brown. Harvard University Press. doi:10.1002/mar.4220020104.

———. 1974b. *Frame Analysis*. Ed. Keith Brown. Harvard University Press. doi:10.1002/mar.4220020104. http://digilib.bc.edu/reserves/en765/sofe/en76506.pdf.

Hall, Edward T. 1966. *The Hidden Dimension*. Ed. Anchor Books Doubleday. *The Hidden Dimension*. Vol. 6. Doubleday. http://www.bookforfree.org/pdf-download/The-Hidden-Dimension-BY-Edward-T.-Hall-ID3764.pdf.

Hardwicke, Catherine. 2008. *Twilight*. Summit Distribution. http://www.twilightthemovie.com/.

Harrigan, Pat, and Noah Wardrip-Fruin, eds. 2007. *Second Person: Role-Playing and Story in Games and Playable Media*. The MIT Press. http://books.google.com/books?id=gf4DQgAACAAJ&pgis=1.

Harris, Helen, Jeremy N Bailenson, Alexia Nielsen, and Nick Yee. 2009. "The Evolution of Social Behavior over Time in Second Life." *Presence Teleoperators Virtual Environments* 18 (6): 434 - 448. doi:10.1162/pres.18.6.434. http://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=48133805&site=ehost-live.

Hecker, Chris. 2010. SpyParty – A Spy Game About Subtle Behavior. *Pre-release*. http://spyparty.com/.

Hecker, Chris, Steve Rabin, Damian Isla, Stuart Reynolds, and Borut Pfeifer. 2009. The Photoshop of AI: Debating the Structure vs Style Decomposition of Game AI. In *AI Summit of the Game Developers Conference*. San Francisco. https://www.cmpevents.com/GD09/a.asp?option=C&V=11&SessID=8279.

Heider, F. 1982. *The psychology of interpersonal relations*. *Journal of Psychology*. Vol. 21. Lawrence Erlbaum. http://books.google.com/books?hl=en&lr=&id=Zh6TDmayL0AC&pgis=1.

Herskovits, Melville J. 1944. "The Nuer: A Description of the Modes of Livelihood and Political Institutions of a Nilotic People . E. E. Evans-Pritchard." *American Anthropologist* 46 (3): 396-400. doi:10.1525/aa.1944.46.3.02a00160. http://doi.wiley.com/10.1525/aa.1944.46.3.02a00160.

Horswill, Ian. 2011. Toward a Rapid Prototyping Environment for Character Behavior. In *Proceedings of the Seventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 40-46. Stanford, CA.

Huvila, Isto, Kim Holmberg, Stefan Ek, and Gunilla Widén-Wulff. 2010. "Social capital in Second Life." *Online Information Review* 34 (2): 295-316. doi:10.1108/14684521011037007. http://www.emeraldinsight.com/10.1108/14684521011037007.

III, Rouse, and Steve Ogden. 2005. *Game Design : Theory & Practice*. *Design*. Wordware Pub. http://books.google.com.mx/books?id=tGePP1Nu_P8C.

Inman, Chris, Vivian H Wright, and Julia A Hartman. 2010. "Use of Second Life in K-12 and Higher Education : A Review of Research." *Review Literature And Arts Of The Americas* 9 (1): 44-63. http://www.ncolr.org/jiol/issues/viewarticle.cfm?volID=9&IssueID=28&ArticleID=143&Source=2.

Isla, Damian. 2005. "Handling Complexity in the Halo 2 AI." *Game Developers Conference*: 12. http://www.gamasutra.com/gdc2005/features/20050311/isla_01.shtml.

Johnson, W. Lewis, Carole Beal, Anna Fowles-Winkler, Ursula Lauper, Stacy Marsella, Shrikanth Narayanan, Dimitra Papachristou, and Hannes Vilhjálmsson. 2004. Tactical Language Training System: An Interim Report. In *7th International Conference on Intelligent Tutoring Systems*, 336-345.

Juul, Jesper. 2005. Half-Real: The Interplay between Game Rules and Game Fiction. In *DiGRA 2005 Conference Changing ViewsWorlds in Play*. http://www.digra.org/dl/.

King, Michael Patric. 2001. "The Agony and the 'Ex'-tacy." *Sex and the City*, June 3.

Kriegel, Michael, MY Lim, Ruth Aylett, Karin Leichtenstern, Lynne Hall, and Paola Rizzo. 2010. A Case Study In Multimodal Interaction Design For Autonomous Game Characters. In *Multimodal Output Generation (MOG) workshop at International Natural Language Generation Conference (INLG*. http://www.ctit.utwente.nl/library/proceedings/wp10-02.pdf#page=23.

Krugman, Paul. 2007. "Innovating Our Way to Financial Crisis ." *New York Times*. http://www.nytimes.com/2007/12/03/opinion/03krugman.html?_r=1&ex=1354338000.

Langton, Christopher G. 1995. "Artificial Life: An Overview" (January 1). http://portal.acm.org/citation.cfm?id=527171.

League, Levo. 2012. Imagine a World Without Innovation in: Literature. *Forbes*. http://www.forbes.com/sites/levoleague/2012/02/27/imagine-a-world-without-innovation-in-literature/.

Lebowitz, M. 1984. "Creating characters in a story-telling universe." *Poetics* 13 (3) (June): 171-194.

Li, Nan, David J Stracuzzi, Gary Cleveland, Pat Langley, Tolga Konik, Dan Shapiro, Kamal Ali, Matthew Molineaux, and David W Aha. 2009. “Constructing game agents from video of human behavior.” *AIIDE09* 2 (3). http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Constructing+Game+Agents+from+Video+of+Human+Behavior#0.

Likert, R. 1932. “A technique for the measurement of attitudes.” *Archives of Psychology* 140 (140): 1-55. doi:10.1111/j.1540-5834.2010.00585.x. http://www.citeulike.org/user/robertlischke/article/2731047.

Lim, Mei Yii, João Dias, Ruth Aylett, and Ana Paiva. 2008. “Improving Adaptiveness in Autonomous Characters.” Ed. Helmut Prendinger, James Lester, and Mitsuru Ishizuka. *Intelligent Virtual Agents Proceedings* 5208: 348–355. http://www.springerlink.com/content/u2602h1854t103m4/.

Lindon Lab Inc. 2003. Second Life. http://secondlife.com.

Loyall, Bryan A. 1997. Believable Agents: Building Interactive Personalities. Carnegie Mellon University.

Lévi-Strauss, Claude. 1966. *The savage mind*. *North*. Vol. 17. University of Chicago Press. doi:10.1007/BF01248674. http://books.google.com/books?id=JI6GVFbP9hAC&pgis=1.

Malinowski, Bronislaw. 2012. *The Sexual Life of Savages In North-Western Melanesia; An Ethnographic Account of Courtship, Marriage and Family Life Among the Natives of the Trobriand Islands, British New Guinea*. Martino Fine Books. http://www.amazon.com/North-Western-Melanesia-Ethnographic-Courtship-Trobriand/dp/1614272425.

Marsella, Stacy, and Jonathan Gratch. 2009. “EMA: A process model of appraisal dynamics.” *Cognitive Systems Research* 10 (1): 70–90. http://linkinghub.elsevier.com/retrieve/pii/S1389041708000314.

Marsella, Stacy, and David Pynadath. 2005. Modeling influence and theory of mind. In *Artificial Intelligence and the Simulation of Behavior*.

Marsella, Stacy, David Pynadath, and Stephen Read. 2004. PsychSim: Agent-based modeling of social interactions and influence. In *Proceedings the International Conference on Cognitive Modeling*. Pittsburg.

Mascarenhas, Samuel, Joao Dias, Rui Prada, and Ana Paiva. 2010. “A Dimensional Model for Cultural Behavior in Virtual Agents.” *Applied Artificial Intelligence* 24 (6) (July): 552-574. doi:10.1080/08839514.2010.492163. http://www.informaworld.com/openurl?genre=article&doi=10.1080/08839514.2010.492163&magic=crossref||D404A21C5BB053405B1A640AFFD44AE3.

Mascarenhas, Samuel, João Dias, Nuno Afonso, Sibylle Enz, Ana Paiva Inesc-id, Av Prof, and Cavaco Silva. 2009. Using Rituals to Express Cultural Differences in Synthetic

Characters. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS09)*, 305-312.

Mateas, Michael. 2002. "Interactive drama, art and artificial intelligence." http://www.cs.cmu.edu/~burks/ThesisSummaries/mateas_summary.pdf.

———. 2003a. "Expressive AI: games and artificial intelligence." *Proceedings of International DiGRA Conference*. http://steel.lcc.gatech.edu/~mateas/publications/MateasDIGRA2003.pdf.

———. 2003b. Expressive AI: Games and Artificial Intelligence. In .

———. 2011. Games Research - The Quest for Procedural Everything. In *Iventing the Future of Games*. Milpitas, CA. http://games.soe.ucsc.edu/future2011.

Mateas, Michael, and Andrew Stern. 2002. "A behavior language for story-based believable agents." *IEEE Intelligent Systems* 17 (4) (July): 39-47. doi:10.1109/MIS.2002.1024751. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1024751.

———. 2004. "A Behavior Language: Joint Action and Behavioral Idioms." Ed. Helmut Prendinger and Mitsuru Ishizuka. *Lifelike Characters Tools Affective Functions and Applications* 194: 1-28. http://books.google.com/books?hl=en&lr=&id=DIKLF1zoW4gC&oi=fnd&pg=PA135&dq=A+Behavior+Language+:+Joint+Action+and+Behavioral+Idioms&ots=LqnXnfqQmO&sig=OzzqWMi84DzS4SZUDwDq7fwlQcY.

———. 2005. Structuring Content in the Façade Interactive Drama Architecture. In *Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2005)*. Vol. 3. Marina del Rey, CA.

McCoy, Josh, and Michael Mateas. 2008. An Integrated Agent for Playing Real-Time Strategy Games. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI 2008)*. Chicago, IL. http://www.aaai.org/Papers/AAAI/2008/AAAI08-208.pdf.

———. 2009. The Computation of Self in Everyday Life : A Dramaturgical Approach for Socially Competent Agents. In *Proceedings of the AAAI Intelligent Narrative Technologies 2 Symposium (AAAI-INT2 2009)*. Stanford, CA.

McCoy, Josh, Michael Mateas, and Noah Wardrip-fruin. 2009. Comme il Faut : A System for Simulating Social Games Between Autonomous Characters. In *Proccedings of the 8th Digital Art and Culture Confernce (DAC 2009)*. Irvine, CA.

McCoy, Josh, Mike Treanor, Ben Samuel, Aaron Reed, Noah Wardrip-fruin, and Michael Mateas. 2012. Prom Week. Center for Games and Playable Media. http://promweekgame.com.

McCoy, Josh, Mike Treanor, Ben Samuel, Brandon Tearse, Michael Mateas, and Noah Wardrip-Fruin. 2010. Authoring Game-based Interactive Narrative using Social Games and Comme il Faut. In *Proceedings of the 4th International Conference & Festival of the Electronic Literature Organization: Archive & Innovate*. Providence, Rhode Island.

McCoy, Josh, Mike Treanor, Ben Samuel, Brandon Tearse, Michael Mateas, and Noah Wardrip-fruin. 2010. Comme il Faut 2 : A fully realized model for socially-oriented gameplay. In *Proceedings of Foundations of Digital Games (FDG 2010) Intelligent Narrative Technologies III Workshop (INT3)*. Monterey, California.

McCrae, R R, and O P John. 1992. "An introduction to the five-factor model and its applications." Ed. Federal Government. *Journal of Personality* 60 (2): 175-215. http://www.ncbi.nlm.nih.gov/pubmed/1635039.

Mead, Margaret. 1928. Coming of age in Samoa: a psychological study of primitive youth for Western civilisation. *OF PRIMITIVE YOUTH FOR WESTERN CIVILIZATION*. William Morrow and Co.

Meehan, James. 1976. *The metanovel : writing stories by computer*. [New Haven]: Yale University Department of Computer Science. http://www.worldcat.org/title/metanovel-writing-stories-by-computer/oclc/606011843.

De Munck, Victor C. 2000. Handbook of Methods in Cultural Anthropology:Handbook of Methods in Cultural Anthropology. Ed. G K Garbett. *American Anthropologist*. doi:10.2307/2801206. http://www.anthrosource.net/doi/abs/10.1525/aa.2000.102.1.183.

Murray, Janet Horowitz. 1997. *Hamlet on the holodeck: the future of narrative in cyberspace*. New York: Free Press. http://books.google.com/books?id=bzmSLtnMZJsC&pgis=1.

Nutt, Christian. 2011. *Rod Humble On Ethical Game Design And The Challenge Of Free Will*. [Online] Gamasutra. Available at http://www.gamasutra.com/view/news/34219/Rod_Humble_Ethical_Game_Design_And_The_Challenge_Of_Free_Will.php, April.

———. 2012. "Beyond Heavy Rain: David Cage on Interactive Narrative." *Gamasutra*. http://www.gamasutra.com/view/feature/171004/beyond_heavy_rain_david_cage_on_.php.

Orkin, Jeff, and Deb Roy. 2007. "The restaurant game: Learning social behavior and language from thousands of players online." *Journal of Game Development* 3 (December): 39-60. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.67.7135&rep=rep1&type=pdf.

Orkin, Jeff, Tynan Smith, Hilke Reckman, and Deb Roy. 2010. "Semi-automatic task recognition for interactive narratives with EAT & RUN." *International Conference On The Foundations Of Digital Games*. http://portal.acm.org/citation.cfm?id=1822309.1822312.

Ortony, Andrew, Gerald L Clore, and Allan Collins. 1990. *The Cognitive Structure of Emotions*. *booksgooglecom*. Cambridge University Press. http://www.amazon.com/dp/0521386640.

de Pascale, Maurizio, Sara Mulatto, and Domenico Prattichizzo. 2008. Bringing haptics to Second Life. In *Proceedings of the 2008 Ambi-Sys Workshop on Haptic User Interfaces in Ambient Media Systems*, 6. February 14. http://dl.acm.org/citation.cfm?id=1413907.1413913.

Pepper, Stephen C. 1961. *World Hypotheses: A Study in Evidence*. University of California Press. http://www.amazon.com/World-Hypotheses-A-Study-Evidence/dp/0520009940.

Perry, David, and Rusel DeMaria. 2009. *David Perry on Game Design: A Brainstorming ToolBox*. Charles River Media. http://www.amazon.com/David-Perry-Game-Design-Brainstorming/dp/1584506687.

Prada, Rui, and Ana Paiva. 2005a. Synthetic group dynamics in entertainment scenarios: creating believable interactions in groups of synthetic characters. In *ACE 05 Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, 334-334:3. ACM. doi:10.1145/1178477.1178585.

———. 2005b. "Believable groups of synthetic characters." *AAMAS 05 Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*: 37-43. doi:10.1145/1082473.1082479. http://portal.acm.org/citation.cfm?doid=1082473.1082479.

———. 2009. "Teaming up humans with autonomous synthetic characters." *AI Communications* 173 (1) (January): 80-103. doi:10.1016/j.artint.2008.08.006. http://linkinghub.elsevier.com/retrieve/pii/S0004370208001161.

Premack, D, and G Woodruff. 1978. "Chimpanzee problem-solving: a test for comprehension." *Science* 202 (4367): 532-535. http://www.ncbi.nlm.nih.gov/pubmed/705342.

Purho, Petri. 2009. Crayon Physics Deluxe. [PC Computer, Apple IPad], Hudson Soft.

Pynadath, David, and Stacy Marsella. 2004. Fitting and Compilation of Multiagent Models through Piecewise Linear Functions. In *Proceedings of the International Conference on Autonomous Agents and Multi Agent Systems (AAMAS04)*, 47:277-295. September. doi:10.1016/j.csda.2003.11.003.

———. 2005. Psychsim: Modeling theory of mind with decision-theoretic agents. In *International Joint Conference on Artificial Intelligence*, 19:1181. Citeseer. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.83.2996&rep=rep1&type=pdf.

Quantic Dream. 2010. Heavy Rain. Sony Computer Entertainment. http://www.heavyrainps3.com.

Rao, A S, and M P Georgeff. 1995. BDI Agents: From Theory to Practice. In *System*, ed. Victor Lesser, 312-319. San Francisco. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.37.7970.

Reiss, Steven. 2008. *The Normal Personality: A New Way of Thinking About People*. New York: Cambridge University Press.

Riedl, Mark, and Michael Young. 2004. An Intent-Driven Planner for Multi-Agent Story Generation. In *Proceedings of Autonomous Agents and Multiagent Systems (AAMAS04)*.

Rollings, Andrew, and Ernest Adams. 2003. *Andrew Rollings and Ernest Adams on Game Design*. New Riders Games. http://www.amazon.com/Andrew-Rollings-Ernest-Adams-Design/dp/1592730019.

Sacks, Harvey. 1974. An analysis of the course of a joke's telling in conversation. In *Explorations in the Ethnography of Speaking*, ed. Richard Bauman and Joel Sherzer, 337-353. Cambridge University Press. http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:An+analysis+of+the+course+of+a+joke's+telling+in+conversation#0.

Salen, K, and E Zimmerman. 2001. *Rules of Play: Game Design Fundamentals*. MIT Press.

Sali, Serdar. 2012. Playing With Words: From Intuition To Evaluation Of Game Dialogue Interfaces. UC Santa Cruz.

Sali, Serdar, and Michael Mateas. 2011. Using Information Visualization to Understand Interactive Narrative: A Case Study on Façade. In *Proceedings of the Fourth International Conference on Interactive Digital Storytelling.* Vol. 7069. Vancouver, Canada. doi:10.1007/978-3-642-25289-1.

Sander, Florian N. 2008. "Linear Interactive Storytelling by." *Drama*.

Schank, Roger. 1975. *Conceptual Information Processing*. New York: Elsevier.

Schank, Roger, and Robert Abelson. 1977. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. L. Erlbaum Associates.

Sengers, Phoebe. 2000. Narrative Intelligence. In , ed. Kerstin Dautenhahn. John Benjamins Publishing Company.

Si, Mei, and S.C. Marsella. 2010. Modeling Rich Characters in Interactive Narrative Games. In *GAMEON-ASIA*. http://people.ict.usc.edu/~meisi/publication/021010gameonasia-final.pdf.

Si, Mei, S.C. Marsella, and D.V. Pynadath. 2010a. "Evaluating Directorial Control in a Character-Centric Interactive Narrative Framework." *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*. http://www.ifaamas.org/Proceedings/aamas2010/pdf/01 Full Papers/26_05_FP_0727.pdf.

Si, Mei, Stacy Marsella, and David Pynadath. 2005a. Thespian: An architecture for interactive pedagogical drama. In *International Conference on Artificial Intelligence in Education*. Citeseer.

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.86.2901&rep=rep1&type=pdf.

———. 2005b. Thespian: Using Multi-Agent Fitting to Craft Interactive Drama. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems - AAMAS '05*, 21. Utrecht, Netherlands: ACM Press.

———. 2007. Proactive authoring for interactive drama: An author's assistant. In *Intelligent Virtual Agents*, 225–237. Springer. http://www.springerlink.com/index/b461w25111812435.pdf.

———. 2010b. "Directorial Control in a Decision-Theoretic Framework for Interactive Narrative." *AAMAS*.

Si, Mei, Stacy Marsella, and David V. Pynadath. 2009a. "Modeling appraisal in theory of mind reasoning." *Autonomous Agents and Multi-Agent Systems* 20 (1) (May): 14-31. doi:10.1007/s10458-009-9093-x. http://www.springerlink.com/index/10.1007/s10458-009-9093-x.

———. 2009b. "Modeling appraisal in theory of mind reasoning." *Autonomous Agents and Multi-Agent Systems* 20 (1) (May): 14-31. doi:10.1007/s10458-009-9093-x. http://www.springerlink.com/index/10.1007/s10458-009-9093-x.

Si, Mei, Stacy Marsella, and David Pynadath. 2006. "Social Norms Models in Thespian: Using Decision Theoretical Framework for Interactive Dramas." *Proceedings AISB* 39. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.85.9655&rep=rep1&type=pdf.

Sims, Roderick. 1999. Interactivity or Narrative? A Critical Analysis of their Impact on Interactive Learning. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 1999*, 627-638.

Skorupski, James. 2009. Storyboard authoring of plan-based interactive dramas. In *Proceedings of the 4th International Conference on Foundations of Digital Games - FDG '09*, 349. ACM Press. doi:10.1145/1536513.1536579. http://portal.acm.org/citation.cfm?doid=1536513.1536579.

Skorupski, James, and Michael Mateas. 2010. Novice-friendly Authoring of Plan-based Interactive Storyboards. In *Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2010)*. Stanford, CA.

Skorupski, James, and Josh McCoy. 2012. *Evaluating Realism for Virtual Supporting Characters*. Santa Cruz, CA. https://www.soe.ucsc.edu/research/technical-reports/ucsc-soe-12-04.

Spierling, Ulrike, Nicolas Szilas, Rui Figueiredo, Antonio Brisson, Ruth Aylett, and Ana Paiva. 2008. *Interactive Storytelling*. Ed. Ulrike Spierling and Nicolas Szilas. *Interactive*

Storytelling. Vol. 5334. Springer Berlin Heidelberg. doi:10.1007/978-3-540-89454-4. http://www.springerlink.com/content/x066lu2010676736/.
Star, Darren. 1998. Sex and the City.
Sullivan, Anne, April Grow, Tabitha Chirrick, Max Stokols, Noah Wardrip-fruin, and Michael Mateas. 2011. Extending CRPGs as an Interactive Storytelling Form. In *Interactive Storytelling 4th International Conference on International Digital Storytelling*, ed. Mei Si, David Thue, Elizabeth Andre, James Lester, Joshua Tanenbaum, and Veronica Zammitto, 164-169. Springer Berlin Heidelberg.
Swanson, Reid, and Andrew Gordon. 2010. A Data-Driven Case-Based Reasoning Approach to Interactive Storytelling. In *The Third International Conference on Interactive Digital Storytelling (ICIDS 2010)*. Edinburgh, UK: Citeseer. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.172.7806&rep=rep1&type=pdf.
Tanenbaum, Joshua. 2011. Imagining New Design Spaces for Interactive Digital Storytelling. In *Interactive Storytelling 4th International Conference on International Digital Storytelling (2011)*, 164-169. Springer Berlin Heidelberg.
The Sims Studio. 2009. The Sims 3. Electronic Arts.
Toy, M, G Wichman, K Arnold, and J. Lane. 1980. *Rogue*. [PC Game].
Trabasso, T., and P. van Den Broek. 1985. "Causal thinking and the representation of Narrative Events." *Journal of Memory and Language* 24 (5) (October): 612-630. doi:10.1016/0749-596X(85)90049-X. http://dx.doi.org/10.1016/0749-596X(85)90049-X.
Turing, A M. 1950. "Computing Machinery and Intelligence." *Mind* 59 (236): 433-460. doi:10.1093/mind/LIX.236.433. http://mind.oxfordjournals.org/cgi/doi/10.1093/mind/LIX.236.433.
Turner, Scott. 1994. *The Creative Process: A Computer Model of Storytelling and Creativity*. Psychology Press. http://www.citeulike.org/group/4585/article/8010058.
Vorderer, Peter, and Jennings Bryant. 2006. *Playing video games: motives, responses, and consequences*. Lawrence Erlbaum Associates. http://books.google.com/books?hl=en&lr=&id=8sY3z98pwM8C&pgis=1.
Wardrip-Fruin, Noah. 2009. *Expressive Processing: Digital Fictions, Computer Games, and Software Studies*. Ed. Matthew Fuller, Lev Manovich, and Noah Wardrip-Fruin. First Edit. Cambridge, Massachusetts: The MIT Press.
Waters, Mark. 2004. *Mean Girls*. Paramount Pictures. http://www.meangirlsmovie.com/.
Weitnauer, Erik, Nick M Thomas, Felix Rabe, and Stefan Kopp. 2008. "Intelligent Agents Living in Social Virtual Environments – Bringing Max Into Second Life." *Germania*: 552-553.

Wiley, Norbert. 1994. *The Semiotic Self*. Chicago, IL: University of Chicago Press. http://www.wiley.com/WileyCDA/WileyTitle/productCd-0745615031.html.

Wiseman, Rosalind. 2003. *Queen Bees and Wannabes: Helping Your Daughter Survive Cliques, Gossip, Boyfriends, and Other Realities of Adolescence*. Three Rivers Press. http://www.amazon.com/Queen-Bees-Wannabes-Boyfriends-Adolescence/dp/1400047927.

Yee, N, H Harris, M Jabon, and J N Bailenson. 2010. "The Expression of Personality in Virtual Worlds." *Social Psychological and Personality Science* 2 (1): 5-12. doi:10.1177/1948550610379056. http://spp.sagepub.com/cgi/doi/10.1177/1948550610379056.

Yee, Nick, and Jeremy N Bailenson. 2008. "A method for longitudinal data collection in Second Life." *Presence Teleoperators Virtual Environments* 17 (6): 1-6. http://libezproxy.open.ac.uk/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=a9h&AN=35003518&site=ehost-live&scope=site.

Yee, Nick, Jeremy N Bailenson, Mark Urbanek, Francis Chang, and Dan Merget. 2007. "The unbearable likeness of being digital: the persistence of nonverbal social norms in online virtual environments." *Cyberpsychology and Behavior* 10 (1): 115-121. http://www.ncbi.nlm.nih.gov/pubmed/17305457.

Yee, Nick, Nicolas Ducheneaut, Mike Yao, and Les Nelson. 2011. "Do Men Heal More When in Drag ? Conflicting Identity Cues Between User and Avatar." *Proteus*: 773-776.

Young, R Michael, and Mark Riedl. 2003. Towards an architecture for intelligent control of narrative in interactive virtual worlds. In *Proceedings of the 8th International Conference on Intelligent User Interfaces*, 310-312. ACM. doi:10.1145/604045.604108. http://portal.acm.org/citation.cfm?id=604045.604108.

Zebrowitz, Leslie A. 2009. "The Affordances of Immersive Virtual Environment Technology for Studying Social Affordances." *Psychological Inquiry* 13 (2): 143-145.