Social Story Worlds With Comme il Faut

Joshua McCoy, Mike Treanor, Ben Samuel, Aaron A. Reed, Michael Mateas, and Noah Wardrip-Fruin

***Abstract*—This paper presents Comme il Faut (CiF), an artificial intelligence system that matches character performances to appropriate social context, with the goal of enabling authors to write high-level rules governing expected character behavior in given social situations, rather than specific fixed choice points in a curated narrative structure. CiF models characters with a complex set of traits, feelings, and relationships, who can form intents, take actions, relate to a shared cultural space, and remember and refer to past events. A set of authored rules encoding appropriate behavior within a specific story world allow these characters to select actions to take (and respond to actions by others) in a manner consistent with their own personal and social concerns as well as a shifting interpersonal context. Through the development and release of *Prom Week*, a complete game using CiF as its narrative engine, we show how the system successfully creates complex narratives that are unique for each player and directed by those players' attempts to make progress towards story goals. We also show how CiF continues to be used in several in-progress interactive experiences (Mismanor and IMMERSE), speaking to the utility and flexibility of its design.**

***Index Terms*—Artificial intelligence, emergent narrative, game design, interactive drama.**

I. Introduction

VIDEO games present a tension between storytelling and player interaction that is not present in most forms of media. While other media are static and tell preauthored stories, the interactivity of video games affords the telling of dynamic stories—collaborations between the game designer and the player. The interactivity of stories told in video games can potentially range from static stories (much like those told in other forms of media) to completely dynamic stories that are procedurally authored [1].

Mainstream state-of-the-art storytelling games, such as *Final Fantasy XIII* [2] and *Heavy Rain* [3], do not offer many options for the player to influence the story, making them more akin to static rather than fully interactive stories. When present, the player's influence on the story is limited to a more local story impact. These games employ the "beads on a string" model of interactive narrative [4], which links sequences of narratively motivated gameplay into a linear order, collapsing and eliminating most consequences of player choice each time the next "bead" is reached. Such designs are common because allowing for branching structures at discrete player choice points otherwise creates an exponentially increasing authorial burden, meaning designers tend to avoid structures that allow for real choice, to the detriment of meaningful player agency within an interactive narrative.

Increasing the impact a player has on the story with current methods is problematic. For an interaction to be meaningful, a decision needs to be made by the player that has some impact on the story world; if the player had no impact on the story world, it would not truly be an interactive experience. Every authored point of player interaction increases the number of potential stories. In effect, authoring an interactive story game is authoring a space of possible stories, and playing such a game is exploring one distinct story of the many possible that could be experienced. The key problem is creating a space of stories while ensuring the quality and consistency of each story.

At present, one of the few alternatives to the "beads on a string model" for increasing player impact on a story is a simple brute force approach. Maintaining story quality while accounting for player impact in these large story spaces is accomplished by manually authoring content for every possible state the player could drive the story into. The fallout of this method of creation can be seen in the massive amounts of content authoring needed to realize the story experience of *Star Wars: The Old Republic* [5]. Even with massive authoring efforts, the best current story games still have a weak coupling of narrative with player choice. In this and other games, players are afforded a large amount of play in the spaces of physical interaction and combat while having very little ability to play with the story.

The power of combat and physical interaction in games comes from how their domains are modeled; instead of accounting for every possible state discretely, they are computational models complete with general rules for their domains. They allow for players to interact with a large space of possibilities with the computational model maintaining consistency and causality within that space, such as the platformer physics of *Super Mario Galaxy 2* [6] or the portal gun mechanics of *Portal 2* [7]. Instead of providing a scripted set of choices, the players are provided with a space of play. To make interactive stories live up to their name, the playability in story games needs to approach that of physical interaction and combat.

Comme il Faut (CiF) [8] is a computational model of social interaction that enables a new class of interactive stories outside the purview of exploration and combat. CiF provides playability by focusing on social interaction; the player's actions have deep impact on the social world and can greatly influence the social future of characters in a CiF-powered game. Social relationships

Manuscript received October 06, 2012; revised June 26, 2013; accepted September 09, 2013. Date of publication February 05, 2014; date of current version June 12, 2014. This work was supported in part by the National Science Foundation (NSF) under Award IIS-0747522 and by an NSF Graduate Research Fellowship.

J. McCoy, B. Samuel, A. A. Reed, M. Mateas, and N. Wardrip-Fruin are with the Computer Science Department, University of California Santa Cruz, Santa Cruz, CA 95064 USA (e-mail: mccoyjo@soe.ucsc.edu).

M. Treanor was with the Computer Science Department, University of California Santa Cruz, Santa Cruz, CA 95064 USA. He is now with the American University, Washington, DC 20016 USA

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCIAIG.2014.2304692

are critical to many stories, and are involved in nearly every story [9], which makes CiF an important step toward making stories more interactive.

CiF is inspired by social science ideas, including dramaturgical analysis [10], in its interactive, authorable model of social interaction for autonomous agents. A primary responsibility for CiF is to retarget patterns of social behavior amidst shifting contexts and to modify character performances to be appropriate to the social context. Social exchanges, defined as multicharacter social interactions whose function is to modify the social state existing within and across the participants, are the structures that are retargeted.

Through the use of social exchanges along with additional encoded social context, CiF increases the impact and interactivity a player has on and with the social aspects of an interactive story. Context is encoded when authors 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 comprises 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 immediately used by CiF. When the rules are used in conjunction with social exchanges, the character behaviors generated by CiF can be rich and surprising.

In this paper, we contribute a detailed description of the structures with which CiF represents social knowledge and how this knowledge is employed to simulate social interactions between characters in a story world. We provide concrete examples of how CiF can be used to enable social character behavior for interactive storytelling in a way that is tractable to the author and flexible for the player. We describe several games which are powered by CiF with an emphasis on CiF's flagship title *Prom Week*. We also present an evaluation of the narratives assembled jointly by players and CiF for *Prom Week,* as well as covering Prom Week's positive critical reception. This evaluation examines both the uniqueness of players' paths through *Prom Week* and how well players achieved their story goals.

II. Related Work

Narrative generation systems [11]–[15] often model storytelling with general mechanisms, rather than focusing on the intricacies of specific domains. In comparison, CiF does not provide a general mechanism for encoding many domains. 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 used by a narrative generation system. Similar goals have been attempted through analysis of crowd-sourced data to discover common play interaction patterns [16] but our approach is fundamentally different, driven by a rules-driven AI system rather than pattern matching from a player-generated corpus.

There are many systems in the domain of modeling interactions between characters or virtual humans based on cognitive or psychological models that reason over competing capacities of a prescribed set of desires [17]–[20]. CiF is an implementation of an alternate, norms-based vision of modeling what characters should be doing. This approach gives characters the affordance to reason over what desires are appropriate for the situation and then to negotiate between those relevant desires [21]. One advantage of this approach is that rather than authoring each scenario in isolation, narrative content can be created based around general social norms that are reusable whenever that pattern of social behavior comes up.

In comparison to hierarchical task networks [22], [23] and behavior trees [24], 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) preconditions or postconditions, CiF chooses characters' behaviors based on all applicable rules in a large rule base that encodes normal social behavior authored for a particular story world.

The Sims 3 is an example of a culturally influential and commercially successful video game that has a highly dynamic social space [25]. Its characters, known as Sims, have traits and desires that inform the social practices (social norms and cluster of expectations) they perform [26]. Two major differences between the systems are 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 the social exchange can be referenced (x is cheating on y if x and y are dating and there is a character z also dating x) where *The Sims 3* can only reference the two characters in an interaction. CiF also allows for both backstory (history of the story world before the player is involved) and play history to be used in reasoning and social exchange performance, a feature completely missing from *The Sims 3*. The richer rules found in CiF allow for each individual authoring effort to be more potent while enabling an entire new set of social reasoning for the characters. While still a rare approach, other more recent systems appearing since CiF's initial design use similar complex reasoning over social states, such as Praxis [27], the engine behind the Versu [28] interactive storytelling platform, and NetworkING [29], which uses a simulated social state to feed a narrative generator.

III. Comme IL Faut

Comme il Faut is a French phrase which translates to "Being in accord with conventions or accepted standards." CiF is a model of social state, a collection of processes which can reason over that social state, and a framework for defining actions which can alter the social state and ways for those actions to be performed. Though CiF is a powerful tool for social reasoning, CiF in and of itself is not a playable experience. Rather, it is intended to be used as a component of a game which wishes to leverage social dynamics; CiF reports what actions characters would like to take, but it is up to the game that is using CiF to interpret how that should be manifested to the player. What follows in this section is an overview of the elements which constitute CiF's representation of characters, the social state,

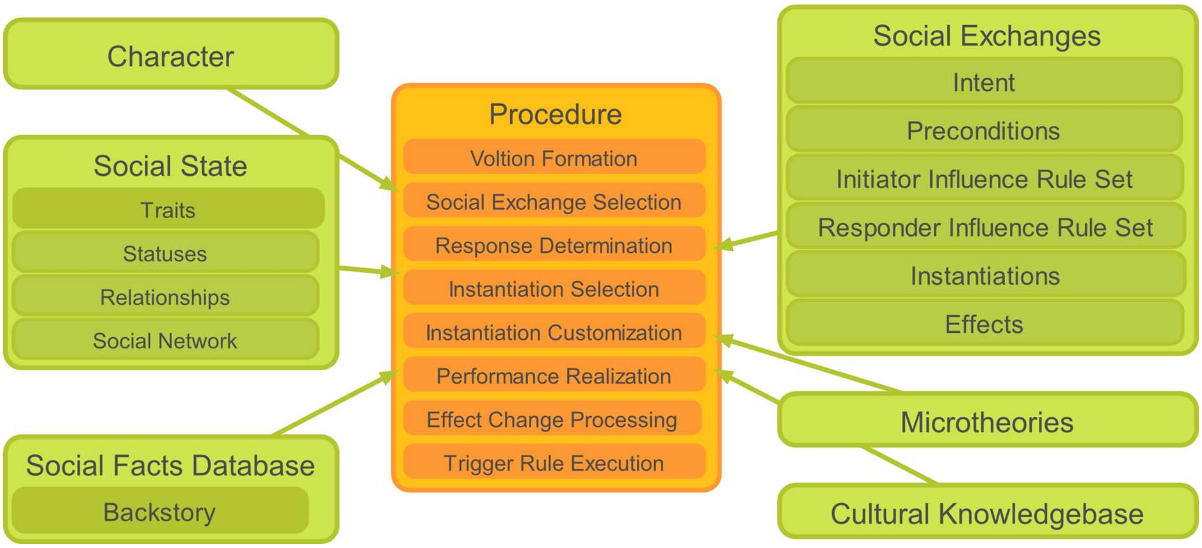

Fig. 1. System architecture diagram of CiF. Characters, the current social state, the history stored in the social facts database, along with authored social exchanges, microtheories, and the cultural knowledge base are used to inform CiF's procedures. Volition formation determines what social exchange characters want to do with one another. After social exchange selection, which is handled by the playable experience leveraging CiF, CiF determines if the responder will accept or reject the intent of the social exchange. The most salient instantiation is selected, and then customized with NLG templates to be consistent with the social state. After presenting the instantiation through performance realization (again, handled by the game using CiF), the effect changes are processed, updating the social state. Finally, trigger rules are executed, which potentially further change the social state, setting the stage for another round of volition formation.

and how characters might interact with one another to change the social state through social exchanges. We also explore the rule system which enables CiF's encoding of social norms, which in turn dictates how characters are inclined to behave toward one another. An overview of the CiF architecture is presented in Fig. 1.

A. Characters

Due to the emphasis in CiF on social norms and how they guide social exchanges, the representation of each character is thin. What makes characters rich and unique is their relational situation in the social world and their interconnected history. This is a direct artifact of the sociological base of CiF; the model of characters is inspired by the concept of semiotic self where the myriad factors of history, experience, future predictions, and social forces define a malleable self that is not lost in larger societal collectives [30]. The system determines the most salient social influences for a character by considering a full context of social norms, history, and current circumstance. But before we cover the complexities of the greater social state, we will first define what constitutes a character in our system. Characters are associated with four primary sets of characteristics: traits, statuses, relationships, and social networks.

B. Social State

1) Traits and Statuses: Traits and statuses are permanent or temporary binary properties, respectively, of a character, which impact how that character performs in the social space. A character might always have traits like brave or intelligent but spend short periods of time with statuses such as depressed or injured. Though structurally similar, by convention traits are immutable, while statuses expire when the conditions that triggered them no longer hold. Statuses can also be directional, so a character might temporarily have the status angry at or infatuated with a second character. Though in traditional narratives personality traits often change over time, CiF's focus on short-form narratives means we do not model this.

2) Relationships and Networks: Relationships are binary states that specify a significant social connection between a pair of characters. For example, two characters might have the relationship of housemates or rivals. Relationships are nonexclusive (a character can have multiple rivals) and nonrestrictive (a character can have a housemate who is also a rival). Relationships work in conjunction with bidirectional, scalar valued social networks. For example, one housemate may have a respect network value of 30 toward her rival, who reciprocates with a respect value of 85.

3) Cultural Knowledge Base: The cultural knowledge base (CKB) is a way to further define the world that CiF-driven characters inhabit, by providing them with a variety of concepts and objects from the story world's cultural context. CiF authors can specify both the items themselves and the ways characters can relate to them, such as desire, ownership, or subjective opinion. For example, the CKB for a CiF game inspired by the cartoon character Garfield [31] might include "Mondays" and "lasagna." If the author had defined the connection types "loves," "hates," and "has," he could set a starting state in which Garfield "hates Mondays," "loves lasagna," and "has lasagna."

In addition, authors can link each item to a single adjective which defines an opinion considered universally true about that object. For instance, the author might say that Mondays are "boring," a cultural construct that all characters agree exists, even if their personal opinion differs. Jon might still "love Mondays" in spite of the perception that they are boring. This complexity of representation opens up a powerful expressive space enabling characters to operate within a cultural context.

C. Social Exchanges

CiF uses the above representations of characters and social state to determine how characters should interact with one another. A social exchange is an attempt by one character to change the social state between him and another character. For example, in a CiF story world about Napoleonic warfare, a general (the initiator) might wish to forge an unlikely alliance by gaining an ally relationship with a commander from the opposing side (the responder). This desire, or volition, of the initiator results from CiF's evaluation of the current social state. The responder will choose to either accept or reject the proposed social exchange based on his own relation to the social state in a process of response determination. For example, some factors that might have influenced the general to seek this alliance could include a mutual love for philosophy (CKB), and his feeling weary of war (status). If the opposing commander accepts the exchange, it may be because the general has recently acted with honor toward him [a social facts database (SFDB) entry; see below] and greatly respects him (network value).

Every social exchange authored for a CiF story world has a single primary intent, or intended change to the social state. Multiple social exchanges for the same intent define narratively distinct ways of achieving the same social outcome. The intents, and thus the social exchanges, a character wants to pursue are recalculated after every social exchange in a process called "volition formation." For each pair of characters, volition formation ranks all possible intents and exchanges based on a hand authored set of social influence rules. Each rule has a weight value which adjusts volition for either a specific social exchange or for an intent (and thus a set of social exchanges) either positively or negatively. (In fact, a majority of rule authoring in CiF involves these specific intents, or microtheories; see Section III-E5.) In the example above, one rule might give a higher weighting to accepting a "start ally" intent if the responder has a high respect network value toward the initiator. Another rule might have a strong negative weight for the same intent if the two characters have traits representing allegiance to opposing sides of the war. As these examples imply, rules are domain specific and in aggregate allow characters to behave appropriately within a specific story world's social context.

Each social exchange is only possible within sensible social contexts. To enforce this, CiF has social exchange preconditions, which specify under which conditions any given social exchange is possible. In our Napoleonic example, a specific "start ally" social exchange might have the precondition that the initiator and the responder are not already allies. Social exchange preconditions forbid certain situations while social influence rules merely change their likelihood. The selection of which to use can be an expressive tool for authors to communicate social norms in the story world. For instance, if two generals from opposing sides can never be allies (because of a hard social exchange precondition) it implies a very different social possibility space than if that situation is unlikely but still possible (because of soft influence rules). One use for preconditions is to distinguish different social exchanges which share the same intent. Starting a friendship between two generals on opposing sides might be an "unlikely alliance" social exchange, while starting a friendship between two soldiers could be a "comrades in arms" social exchange. "Unlikely alliance" and "comrades in arms" share the same intent (become friends) but would have distinct preconditions, and possibly different side effects to the social state (see below).

Additionally, authors might want to provide multiple ways for each individual social exchange to be performed. This not only provides variety, but also can demonstrate subtle nuances between differing social states. These individual performances are called instantiations. Every social exchange can have an arbitrary number of instantiations, and each instantiation narrates a specific instance of a social exchange to the player. Instantiations are intended to be the primary way in which the player learns about changes to the social state. Instantiations have their own set of preconditions, separate from social exchange preconditions. While social exchange preconditions dictate which social exchanges are possible within the current social state, instantiation preconditions determine which instantiations are allowed within the current social exchange. Each instantiation performs either an accept or a reject of the proposed social exchange (see above). Once a social exchange has been selected between two characters, CiF determines which instantiation of that exchange to play in the process of instantiation precondition evaluation. If multiple instantiation preconditions evaluate to true, the most salient instantiation is selected. Salience value is generally correlated to the complexity, or strictness, of the instantiation precondition; the more complex the instantiation precondition, the more social state knowledge can be embedded in the instantiation's authored content.

Instantiations are one of the primary ways we take advantage of retargeting social behavior; any given instantiation can be played between any pair of characters in any social state allowed by both the instantiation preconditions and the preconditions of its corresponding social exchange. The term retargeting is in wide use in computer graphics and animation [32]; we introduce it here to describe a similar concept in the domain of interactive narrative. Although this is extremely powerful, it can also be difficult to author for, as any given instantiation can appear in a potentially large variety of social contexts. Since instantiation and social exchange preconditions are the only social facts the author can guarantee to be true at the time of instantiation performance, the more complex the preconditions, the more specific elements to the social state the author can reference, often resulting in richer performances. Regardless of the complexity of the instantiation chosen, CiF will customize it to ensure it is consistent with the social state at the time of the instantiation's performance. The nature of the performance is determined by the system or game using CiF.

In addition to unique preconditions, every instantiation has its own postconditions or effect changes on the social state. These generally take the form of adjusting network values between the characters involved in the social exchange by a fixed amount, starting or ending relationships, or bestowing or removing statuses. There is no limit to the number of effect changes associated with a particular instantiation, though only characters directly involved in the social exchange are permitted to be directly affected. The social state of other characters may be affected by the social fallout of an exchange through the use of trigger rules (see below).

Typically authors will match up the effect changes of an instantiation with the intent of the social exchange it is affiliated with. Instantiations for the social exchange "unlikely alliance," which has the intent to begin a friendship relationship, will likely at the very least establish a friendship relationship between the initiator and the responder if the instantiation is marked as accepted, and deny the relationship if the instantiation is marked as rejected. However, this is merely a convention, and authors are free to author outside of this convention if deemed appropriate for the stories they want to tell. That said, at minimum, each social exchange should have at least two instantiations: one for the case in which the social exchange is accepted, and one for when the exchange is rejected.

The influence rules, social changes, and instantiation 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.

D. Social History

In fiction, a character's actions are informed not just by the present social situation but by his or her memory of actions in the past taken by both himself and others. CiF represents this with an SFDB that records all actions taken in the story world in a form that can be referenced by both influence rules and instantiation performances. CiF does not simulate hidden information: all events that occur are assumed to be immediately known about by all characters.

To achieve this, instantiations are given three pieces of metadata. The first is the time that social exchange took place. The second is a performance realization string that allows the attached event to be described in natural language from the point of view of any character. By using basic natural language generation (NLG) templates (described in detail in Section III-G), the system can swap out names and pronouns to produce texts like "Alan stole Bill's watch," "you stole my watch," or "I stole your watch." The third piece of instantiation metadata is a set of SFDB labels that place the event in one or more author-defined categories. These make it possible for rules to query about whether certain types of actions have been recently performed, at a granularity defined by the author, and for instantiations to request an example of an action that meets either broad or specific criteria.

For example, in a Napoleonic story world, a character might have "offered insult" to another character, a social exchange with the intent "start rivalry." Perhaps the instantiation chosen has the performance realization string "%i% made scandalous remarks about %rp% mother." %i% means the initiator, and %rp% means the responder's possessive pronoun. At the time the scene is narrated, this text might be realized: "Alphonse made scandalous remarks about Jean-Pierre's mother." This instantiation might also be given the SFDB labels "risque" and "cruel."

Later on, CiF might be considering whether the insulted character (the responder in the above example) is willing to accept an offer of friendship from the person who insulted him. In addition to all the considerations described above related to the current social state, there might be influence rules related to information in the SFDB. For instance, one rule might lower a responder's volition to accept a "start friendship" intent if the initiator has recently done something "cruel" to him. ("Recently" can be defined precisely in the rule: some possible alternatives include "in the past n turns," "m times in the past n turns," or "at any point during this game session.") Furthermore, the existence of this past cruel behavior in the SFDB might influence the selection of an instantiation narrating the responder's rejection in a way which specifically references the cruel event. The SFDB label can be referenced in another NLG template to allow for characters to refer to specific incidents by name. So, for instance, a line of instantiation dialog like "You really expect me to accept your offer after %SFDB_(cruel, i, r, 7)%?" might be realized at runtime as "You really expect me to accept your offer after you made scandalous remarks about my mother?" (The parameters to the NLG template requesting an event from the SFDB are the event's SFDB label, the event's initiator, the event's responder, and the time window, here seven turns.) The result is a dynamically customized line of dialog that inserts a reference to an appropriate event within the current playthrough in an appropriate place.

In addition to events generated at runtime, the SFDB can also be prepopulated with events representing the backstory that was supposed to have happened before the game began. These events look identical to runtime events stored in the SFDB, except with negative time stamps (assuming the first turn of a given playthrough begins counting at 1). This allows instantiations to begin immediately leveraging both rules and instantiations representing past character actions, which allows for introducing significant past events between characters naturally as they come up in dialog narrating present social interactions. As the player builds up an SFDB of more recent events, these begin to take precedence in both rule considerations and dialog references.

E. 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 in the process of rule evaluation. Any of the aforementioned aspects of social state (relationships, traits, networks, CKB, SFDB, etc.) can constitute the left-hand side (condition) of a rule in the form of predicates. Rule conditions can be composed of an arbitrary number of predicates, allowing for rules of varying complexity. Rules are used throughout CiF in numerous ways, several of which have been discussed above; social exchange preconditions, instantiation preconditions, and influence rules are all encoded as CiF rules. Simple rules with few predicates can be used to capture general cases of the social state. For example, characters with high mutual respect network values will be more inclined to start and accept a "become allies" social exchange. Rules with a larger number of more specific predicates represent very specific aspects of the social state, such as a rule that only evaluates to true if a character named Demetrius has been involved with at least three instantiations with the SFDB label "cruel," with a second character with the status "lovestruck" (or even the specific character Helena).

CiF uses rules to reason over the social world when making decisions about social exchanges. The details of rule implementation can be found in [33]. 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.

To add an additional level of utility, CiF allows rules to be created and evaluated externally to its processes. An application that employs CiF can evaluate rules at any time, even the ones created dynamically at runtime. This can be valuable, as it provides a hook for the rich social state of CiF to be accessed in any way the designer wants. For example, in the high school themed CiF-based game *Prom Week* (described below), rules are normally used to determine character volitions and response determination, but they are also used to evaluate each level's goals (e.g., "get Zack a date for prom"). These external rules free game designers to leverage CiF's social state in any way they see fit.

1) Predicates: Predicates are the binding between the current social state as modeled by CiF and the authoring of 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. First is a set of 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 SFDB, and cultural items in the world found in the CKB, which were described in detail above.

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. CiF supports three modes of predicate evaluation: true now, true in history, and times true.

In true now mode, the rule simply uses the current social state at the time of evaluation to determine truth. This is the default evaluation mode for predicates. SFDB predicates cannot be true now by design.

Every predicate other than trait and CKB predicates can be evaluated in the true in history mode. True in history determines if the predicate has been asserted on the right-hand side of an effect change or trigger rule in the past. Though SFDB predicates may seem similar to other predicate types using true in history mode, the latter queries states and state changes (such as shifting statuses or relationships) rather than SFDB labels on social exchanges. SFDB labels are meant to capture an impression of a social exchange by literally associating it with labels such as "diabolic" or "kind." Another difference is evaluation efficiency: since comparing a predicate during rule evaluation to all predicates that have taken effect in the past is expensive, marking a subset of the most commonly searched for history predicates as SFDB labels significantly reduces the search space. Last, the specificity afforded by being able to search the history for particular predicates of any type, not just labels, permits greater authorial expression; for example, an author might want to account for a situation where a character increased their "respect" network value toward another character by 33 within the past four turns, as opposed to querying a general label such as doing something "praiseworthy."

The times true mode determines how many times the predicate is true in the current social state. For example, to get the status of "popular," a character might need to have the "friends" relationship with four or more characters. This predicate mode simplifies writing long rule conditions. For example, take a rule with four predicates:

```
relationship(Friends, x, y) and relationship(Friends, x, z) and relationship(Friends, x, w) and relationship(Friends, x, u)
```

This could be rewritten as a rule with a single "times true" predicate. The predicate author only needs to specify a single "friends" relationship predicate, mark it to use the times true mode of evaluation, and specify the number of times it must be true, in this case four. Times true predicates gain some additional power by permitting the author to specify which character variables should be held static in the character binding process, and which are allowed to change. Making one character static and the other variable allows for evaluating facts about a single character in the social space as in the above example, wherein the author wants to see if a single character has at least four "friends," or if a character has made three "diabolic" social exchanges in general. Making both characters static allows for checking the social state for facts about a specific pair of characters, such as seeing if one character has made three "diabolic" social exchanges with a particular second character. In the above example, we could set the first character variable to be static. CiF would then determine how many characters could be bound to the second character variable to make the predicate evaluate to true. The number of true bindings is then compared to the times true number to finalize the evaluation.

Combining different evaluation types with different predicate types yields 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 within a history window, allowing for characters to know, for example, how many times another character has been "romantic" toward them within the specified time frame. 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 "betrayed" or "abandoned." One could readily imagine a story world in which characters begin to pity, or perhaps superstitiously avoid, a character that has had many bad things happen to them.

2) Influence Rules: 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 alter the desires of the agents to engage in social exchanges. The right-hand side of every rule inside an IRS is a weight that represents how important the rule is in determining

TABLE I
EXAMPLE INFLUENCE RULES FROM *PROM WEEK*

Condition (Left-Hand-Side)	Weight to an Intent (Right-Hand-Side)	Description
`status(CheatingOn,r,i)`	`intent(~relationship(Dating,i,r)) + 15`	If someone is cheating on you, your desire to stop dating them increases.
`relationship(Friends,i,r) && status(AngryAt,i,r)`	`intent(network(Buddy,i,r) +) - 5`	If you are angry with a friend, your desire to do friendly things with them is lessened a little.
`status(HasACrushOn,i,r) && SFDB(Romantic,r,i)`	`intent(Relationship(Friends,i,r)) -10`	If your crush has done something romantic to you recently, your desire to become friends with them goes down (in favor of being "more than friends")

intents, where an intent is the intended change in social state after performing a social exchange (e.g., have two characters become friends). Social exchanges have two IRSs, an initiator IRS which determines when a character will (or will not) want to engage in a social exchange, and a responder IRS, which determines whether the responder will accept or reject the intent of the social exchange if someone else tries to engage them. During volition formation, rules throughout every initiator IRS and in all microtheories (see Section III-E5) are considered and their weights tallied—the social exchanges with the highest scored weights represent the social exchanges the initiator most wants to perform. A similar scoring mechanism is used for the responder: microtheories and responder influence rule sets are evaluated and their weights tallied to determine if the responder will accept or reject the social exchange.

Influence rules are CiF rules where the left-hand side is a social condition and the right-hand side consists of a scalar value weight. Influence rules can be associated with specific social exchanges or with microtheories. CiF's processes evaluate influence rules. If the rule evaluates to true, CiF adds the weight value to a character's volition toward either the specific social exchange the influence rule is connected to, or, if the influence rule was given a general intent, the weight is added to the volition of all social exchanges which have that intent. 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. Some influence rules authored for *Prom Week*, the first game that used CiF, can be seen in Table I.

3) Trigger Rules: There are a special set of rules known as trigger rules that are not used in volition formulation. They are fundamentally similar to the effect changes associated with instantiations as they actually change the social state if certain conditions hold true. Unlike effect changes, trigger rules are not associated with a specific instantiation, but exist on a universal scope and can be run at any time, usually after a social exchange has been played.

For example, in a story world where characters can have the dating relationship with each other, it might be useful to know if a character should receive the status "two-timer" by dating multiple people at once. Since there may be several social exchanges, each with their own set of instantiations that lead to "dating", authoring without trigger rules would necessitate checking to see if a character is already "dating" someone in every single instantiation that bestows a "dating" relationship, and if they are, giving them the "two-timer" status. With trigger rules, the author need only write a single rule: if a character is "dating" more than one person, give them the "two-timer" status. Now any instantiations that bestow a "dating" relationship can focus solely on that, and if afterwards a character is "dating" two people, the trigger rule will catch it and mark them a "two-timer." Trigger rules are both an authoring convenience and a useful means of enforcing social definitions (e.g., if monogamy is the expected social norm in this story world, characters who date more than one person should be given the "two-timer" status).

4) Time-Ordered Rules: During the development of CiF, we encountered authoring situations where temporal reasoning was useful, especially in capturing chains of social state change in history. For example, in the *Prom Week* story world, when a character has a second character do something mean to them, and then a third person is mean to the second, the first character should have an increased desire to start a "friends" relationship with the third, since the third essentially defended their honor. This "knight in shining armor" influence rule would be impossible to capture without encoding its chronology. Time-ordered rules are an alternate evaluation mode for rules that allow for this type of temporal evaluation.

The 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. 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). 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. By using time-ordered rules, authors can craft story worlds where the characters react not only to isolated, individual changes to the social state, but also can interpret sequences of actions as entirely new patterns of social interaction, and respond accordingly.

5) Microtheories: While influence rule sets allow for a great deal of power and flexibility, they can become unwieldy and difficult to maintain in complex story worlds and through many revisions. A disadvantage of the "big bag of rules" approach is

TABLE II
TEMPLATES IN CIF'S NLG SYSTEM (* DENOTES A TEMPLATE SPECIFIC TO *PROM WEEK*)

NLG Tags	Examples and Explanations
Roles	**%i% %r% %o%** The name of the character (initiator, responder, other) bound to the role slot.
Role Possessive	**%ip% %rp% %op%** The corresponding character name in its possessive form.
Role (Subject)	**%is% %rs% %os%** The character name as the subject of a sentence (this lets a character refer to himself as "I")
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.

that as the number of rules grows too large for an author to keep track of, redundant, overlapping, or contradictory rules may appear. This problem is only worsened if there are multiple rule authors creating content. To help address this problem, CiF breaks rules into sets called microtheories unified by a precondition.

Each microtheory contains a set of rules applying to one or more predicates on the social state. A buddy cop story world might include a "partner" relation, and a microtheory for the predicate relationship(partner,x,y). The rules in this microtheory can influence the volition of any social exchange and consider any factor in the social state, but they are only consulted when the first character being considered has the "partner" relation with the second. In effect, the microtheory encapsulates the commonsense reasoning for what it means for that predicate to be true in the current story world: in this case, what it means to be partners in a buddy cop story. Some rules within this microtheory might include: partners are likely to accept "need backup" social exchanges from each other; a partner is likely to want to initiate a "get revenge" social exchange on someone who gave his partner the status "injured"; a partner with the trait "loyal" is highly unlikely to accept a "request reassignment" exchange from a "partner," even one who had recently done something marked as "foolish" in the SFDB.

All rules in microtheories are associated with intents. This means that each rule in a microtheory impacts a character's volition to engage in all social exchanges labeled with that rule's intent. 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, if a character x had the status "feels superior to" a character y, it would generally negatively impact x's desire to befriend y, which would be encoded in the status' own microtheory. However, when authoring the initiator IRS of the social exchange "give advice," it is reasonable that x might in fact be more inclined to want to give advice to y if x is feeling superior to them. And, given the right social state, "give advice" is a social exchange that could potentially lead two characters to friendship. This provides a sense as to how microtheories and the influence rule sets interplay and complement each other. More complete technical details about the implementation of microtheories can be found in [33].

F. Performance Script Generation

After an initiator proposes a social exchange and the responder accepts or rejects it, a specific instantiation is chosen based on the current social state and several other factors (including which instantiations have recently been chosen). This instantiation needs to be performed somehow to communicate the social results from the initiator's action. While this performance could in theory take on any form, from character animation to an abstract or iconic representation, a common approach is for each instantiation to contain a set of hand-authored sequential lines of dialog narrating and justifying the specific changes to the social state. Because CiF has so much information about the characters, history, and social state, this dialog can make heavy use of template-based NLG to produce responses that are more customized to the player's unique situation in the story world. We discuss the template-based NLG of *Prom Week* below, but some of the factors that might go into text variation include gender, character-specific slang or dialect, CKB items connected to a character, SFDB references involving a character, the names of characters with specific relations or network values toward a certain character, and so on.

G. NLG Templates

NLG templates are used to customize pieces of authored dialog to the current situation (see Table II). On the simplest level, this lets characters refer to each other by name: "Hello, %r%!" can become "Hello, Thomas!" (the percent signs wrap the template request, which in this case is "r," for the responder being spoken to in this line by the initiator). Templates can also be used to substitute appropriate pronouns, or vocabulary specific to a certain character: the above line could be further customized as "%greeting%, %r%!" where "greeting" might be defined as "What's up" for a casual character and "Good to see you" for a more formal one. On a more complex level, templates can request natural language representations of items from cultural knowledge or SFDBs. A template like %CKB_((i, "likes"), (r, "dislikes"), "lame")% is requesting a cultural reference that the initiator likes, the responder dislikes, and the item is generally considered lame. (It is assumed the author has given this instantiation a precondition specifying that such an item exists in the current social context.) Similarly, requests for references to past actions like %SFDB_("kind," i, o, 20)% are able to use the action's performance realization string to output the correct text (such as "you and I threw Jessica a surprise party") even in complicated situations like this one involving multiple characters, any of whom might be the speaker or the recipient of an utterance. (The "o" in this example refers to "other," the slot for either a third present character or a nonpresent character who is the subject of discussion, and the "20" limits this SFDB lookup to events that happened within the last 20 turns.) With a flexible set of NLG templates used liberally, the particulars of one instantiation performance can vary significantly from one usage to the next, which helps reduce significantly the potential deadening effect from reusing authored content.

IV. Playable Experiences Using CiF

The primary application of CiF in a playable experience is the social simulation game *Prom Week*, which was released on February 14, 2012. Links to play the game for free can be found at promweekgame.com. The following section describes how *Prom Week* made use of CiF to enable a narrative social puzzle game with rich characters. Following this will be an outline of several other projects that have used CiF to varying extent with different game designs and domains.

A. *Prom Week*

Gameplay in *Prom Week* revolves around the social lives of 18 characters at a high school in the week before their senior dance. Inspired by classic high school movies from the past few decades, the game parodies the intense social jockeying and emotional rollercoasters of a memorable week for many new graduates. In any given "story," or a campaign, the player is given a set of goals to potentially complete during the week leading up to the prom. For example, in Zack's story, one goal is to get him a prom date. Goals can be satisfied through an open-ended set of solutions discovered through interaction with the characters and social state. For example, the player could have Zack form a friendship with a popular character over a shared interest, or exploit another character's trait of "competitive" to make an enemy when Zack flirts with someone the competitive character has a crush on.

The player works toward goals by choosing social exchanges for each character to initiate (Fig. 2). The player chooses from the top social exchanges that each character desires to play with each other character. CiF provides this ordered list based on the outcome of the volition formation process.

Because the gameplay of *Prom Week* involves manipulating the social space, which is the primary story content of the kind of high school narrative we wanted to emulate, the gameplay is the story. Every action the player takes advances the game's narrative and sends ripples throughout the internal social state, which in turn affects which actions are available in subsequent turns. CiF is a partner of the player, giving the gameplay narrative meaning and shape. This is in contrast to a sandbox game in which gameplay may be the story, but the story is formed only in the mind of the player, and not understood or reasoned over by the system. While CiF-enabled stories are authored in the sense

Fig. 2. Screenshot of the *Prom Week* interface. Oswald has been selected as the initiator, and Doug is the responder. The far left thought bubble contains all of the social exchanges Oswald wants to do with Doug, the product of volition formation reasoning over the current social state.

that the designers create the initial situation, define the goals for each scenario, create the microtheories, social exchanges and instantiations, it is CiF that enables emergent solutions to each social puzzle, making the resulting story space highly dynamic and responsive to player action.

1) Stories: A player of *Prom Week* begins by selecting a story. A story is a collection of levels, each representing a specific time and place in the week before the prom, where the player can take social actions involving a particular subset of the characters in the story. In addition to the goal of getting Zack a date, other example goals include ending Zack's war against a bully, or getting Zack into a relationship with someone with the status "popular." Goals in a story are sometimes designed to be complementary: ending a rivalry with a "popular" bully could improve Zack's relations with the "popular" crowd, which could help his other goals. Sometimes they are designed to be in direct opposition: a goal of making several friends is mutually exclusive with a goal of making a clean break from high school and ending every friendship. As mentioned above, objectives can be met in a variety of ways: the player could forge a friendship between Zack and the bully, or perhaps make the bully lose his social standing, which might change his antagonism toward Zack.

Story goals are a good example of external rules. They are *Prom Week* specific, but reason over elements of CiF's social state. After every social exchange, *Prom Week* uses CiF's rule evaluation system to check to see if any of the story goals have become true (or, if any that used to be true have become false). If there has been a change, then *Prom Week* lets the player know that story goal progress has been made or lost.

Every story's last level takes place at the prom. After the player runs out of turns, or decides to skip to the end of the night, a customized ending is presented that reflects the combination of goals achieved. For example, Zack's story might happily end with him becoming the prom king if the player was able to get him to date a popular person. Or, if the player had him abandon his unpopular friends to reach this goal, he might get a bittersweet ending where he still becomes prom king, but is confronted by his old friends. Every story has many possible endings for various combinations of goals the player might have completed. As the player finds more endings, additional stories are unlocked.

2) Social Physics: Prom Week allows players to solve goals flexibly, while maintaining consistent and believable characters. CiF enables a style of gameplay we call social physics. While video games have achieved a high level of playability in physical spaces, with activities like combat, movement, and physics-based environmental manipulation all well explored, *Prom Week* set out to make social spaces as playable as physical spaces. The goal was not to recreate the everyday social world, but to create social dynamics specifically crafted for a targeted experience—just as platforming games do not reproduce the physics of the everyday world, but rather an enjoyable simplification tuned for gameplay, and fiction writers portray behavior and dialog in stylized fashions that differ markedly from typical conversation.

Without a system like CiF, representing social interactions between any two characters in our story that take into account cultural context, personal history, and current relationships would be impractical, or perhaps impossible. The space of contexts (states of the virtual world) and social interactions (player interactions) is prohibitively large and not amenable to brute-force authoring. CiF provides knowledge representation and processes that model social interactions to make this ambitious goal tractable to implement.

Prom Week's simulation of social state involves influence rules and microtheories about the following model of a social state:

- relationships (3): friends, dating, and enemies;
- social networks (3): buddy, romance, and cool;
- statuses (34): popular, embarrassed, angry at, pities, cheater, heartbroken, cheerful, confused, lonely, excited, popular, desperate, trusts, has a crush on, anxious, etc.;
- traits (44): competitive, sex magnet, witty, attention hog, brainy, deep, shallow, humble, arrogant, hothead, emotional, self destructive, etc.;
- social fact database labels (13): cool, lame, romantic, failed romance, gross, funny, bad ass, mean, nice, taboo, rude, embarrassing, and misunderstood;
- CKB adjectives (10): cool, lame, romantic, gross, funny, bad ass, mean, nice, taboo, and rude;
- CKB connection types (4): likes, dislikes, wants, and has.

Given the above representation of a social state, over 5000 influence rules were created to represent the social norms of *Prom Week*. The following example illustrates how this model of the social world was used to represent our target of a lighthearted high school drama.

Simon is a character with the traits "oblivious" and "witty." Naomi is a character with the trait "attractive." Simon has the status of "has a crush on" Naomi, and Naomi has the status of "popular." Naomi and Simon have the relationship "friends." Simon has a high "romance network" value toward Naomi but she has a very low "romance network" value toward him. All other network values are neutral. The CKB states that both Simon and Naomi like scientific calculators, which are lame. In the social fact database the past action Simon took toward Naomi marked as "embarrassing" is labeled as "Simon misunderstood Naomi asking for help on homework as a romantic advance."

Because Simon has a crush on Naomi, the influence rules of the "has a crush on" microtheory will increase his desire to ask Naomi on a date. And while the microtheory for "doing embarrassing things" decreases a character's desire to ask someone out, the microtheory for his trait of "oblivious" counteracts the effect. For these reasons and others, the list of social exchanges Simon wants to engage in with Naomi begins with "ask out." When the player chooses to have Simon "ask out" Naomi, CiF determines that she will reject him. One of the reasons for this is that the microtheory for the status "popular" contains influence rules that lower a character's volition to do romantic actions with people who are not popular and especially those who have done embarrassing things recently. Because Simon and Naomi are friends, the particular instantiation chosen involves Naomi kindly letting him down.

B. Other Playable Experiences

Another playable experience created with CiF is Mismanor [34]. Mismanor is a historical, character-driven fantasy story about six people interacting at a country manor. In contrast to *Prom Week*, which used CiF to create a social simulation with a "god's eye" perspective where the player can have any pair of characters play a social exchange with each other, Mismanor creates a first-person experience where the player controls a single character embedded within a specific plot. While all characters were predefined in *Prom Week*, Mismanor lets players choose a set of traits for their character when the experience begins, and restricts available options on each turn to those CiF calculates are likely to be performed by such a character within the current social situation. The player can initiate social exchanges which nonplayer characters can accept or reject, but those characters can also make social moves on the player, who in turn chooses to accept or reject their intent.

Several changes were made to CiF to support a more plot-driven style of gameplay. Items and knowledge were added as first-class entities to the system. While these entities cannot initiate social exchanges, they can be the target of social exchanges designed to interact with them, and store information (through traits or statuses) that can be reasoned over in addition to the other components of the social state. An example item might be a "bottle," which has the traits "drinkable" and "alcohol" and the status "full." A social exchange designed to represent drinking might change this status to "empty," and (because of the "alcohol" trait) give the initiator the status "tipsy." Similarly, knowledge encodes information about significant plot points. A set of statuses like "known by Violet" keeps track of who has learned what information, social exchanges based on sharing information can change these statuses, and microtheories can reason over how a character's behavior might change based on what she knows or does not know. Knowledge might also be tagged with classifier traits like "secret": this enables microtheories describing situations such as gossipy characters being more likely to want to share secrets. The state of knowledge during a particular game could even change at runtime with the addition of statuses like "true" and "false."

Mismanor is a research prototype and has not yet been released as a final game. But it demonstrates the flexibility that CiF allows for creating different flavors of narrative experiences using the same core technology. Information hiding, interaction with objects, and progression through a set of plot points were all relatively easy to add by extending CiF to allow for the new static entities items and knowledge in addition to active characters. Once a framework for social reasoning is in place, it can be used to tell a wide variety of possible stories involving reactive characters.

CiF has also been selected to be a core piece of technology in the IMMERSE project, a Defense Advanced Research Projects Agency (DARPA) experiment designed to teach soldiers good stranger behavior. Rather than teach the idiosyncrasies of a specific culture, IMMERSE employs CiF to create a world that rewards users for picking up on universal social cues, such as recognizing the gestures and postures of others and mirroring them. Unlike the turn-based *Prom Week*, IMMERSE takes place in real time, and the player physically performs social exchanges and otherwise interacts with the world through embodied movement captured with a Kinect. Although the project is still in development, it is heartening to see that CiF can be used in technology with real-world consequences, potentially even saving lives.

In the context of the European research project Social games for conflIct REsolution based on natural iNteraction (SIREN) [35], CiF was used in the design of a training system that would enable children to explore a broad spectrum of conflict resolution strategies in a safe environment. Though the project is no longer actively being developed, CiF was particularly relevant for the system since it explicitly modeled multicharacter social interactions. CiF was being paired with Fearnot AffecTIve Mind Architecture (FAtiMA) [36], an agent architecture that treats emotions as valenced evaluations of the world which affect and are affected by goals in a continuous process. In a conflict resolution training system, CiF would be used to directly encode social dynamics extracted from organizational behavior theory while FAtiMA would have a stronger role in creating dynamic scenarios in which agent attitudes change continuously according to real-time user interactions. The combination would allow exploration of a variety of contextual factors related to conflict.

V. Evaluation

While there are potentially many ways to evaluate a system like CiF, our first evaluations focus on the CiF-based game that has had the most players: *Prom Week*. We begin by presenting a qualitative look at its reception through awards and reviews. Next, we present a quantitative analysis of user-generated gameplay traces. Note that the evaluation presented in this paper is only of popular reception and gameplay traces. A more rigorous evaluation of how well CiF and *Prom Week* fulfill their design goals is future work. Furthermore, developing a method to evaluate the perceived social and narrative qualities of an interactive experience is itself a novel research contribution which we leave for future work.

A. Critical Reception

Even though *Prom Week* permits players to shape their own stories, analyzing game traces fails to convey how satisfying those stories may or may not have been. To address this informally in a qualitative sense, we turn to some of the critical reception and reviews *Prom Week* has received since its release.

Several trusted sources of video game news and reviews [37]–[39] have spoken on both the technical and emotional achievements of *Prom Week*. Game news site *Rock Paper Shotgun*'s reporter confessed that "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." *Play This Thing* called *Prom Week* "…a notable advance in the state of the art of interactive narrative design." Alastair Stephens of the site *Story Wonk* says that "…like all successful stories, [*Prom Week*] swiftly moves beyond the mechanical, beyond the ludic, to the personal and emotional."

Prom Week garnered recognition in competitive settings as well. It was selected as a finalist in the 2012 Independent Games Festival in the category of Technical Excellence, and was also a finalist at the 2012 IndieCade festival. It won the 2012 Intelligent Virtual Agents (IVA) Gathering of Lifelike Agents (GALA) demo and video festival. *Prom Week* was also chosen as AIGameDev's editor's choice for Best AI in an Independent Game in their 2012 Awards for Game AI competition.

It can be difficult to measure the impact a game leaves on its audience. However, early qualitative and quantitative (see below) analysis suggests *Prom Week* has successfully employed innovative technology that enables previously unexplored forms of gameplay and interactive narrative. Players have unique experiences that are driven by story and character and which can produce emotional, meaningful responses in their audience.

B. Data Analysis

The evaluation of *Prom Week* was done using play traces of the game generated by users. Analyzing traces generated from real play situations enables evaluating the impact and interactivity players have on their unfolding stories. Since CiF is the core of *Prom Week*, this analysis also serves as an evaluation of the impact on the story that players are afforded by CiF. Even with the large amount of variation supported by CiF in a story world 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) Play Traces From Prom Week: As players experience *Prom Week*, the system saves their interactions as traces on an external server. These traces provide data for saving and continuing play sessions and contain the information needed to resimulate the social state created by the player. The trace is associated with an anonymous and unique identifier that represents the player and is used to track a player across play sessions.

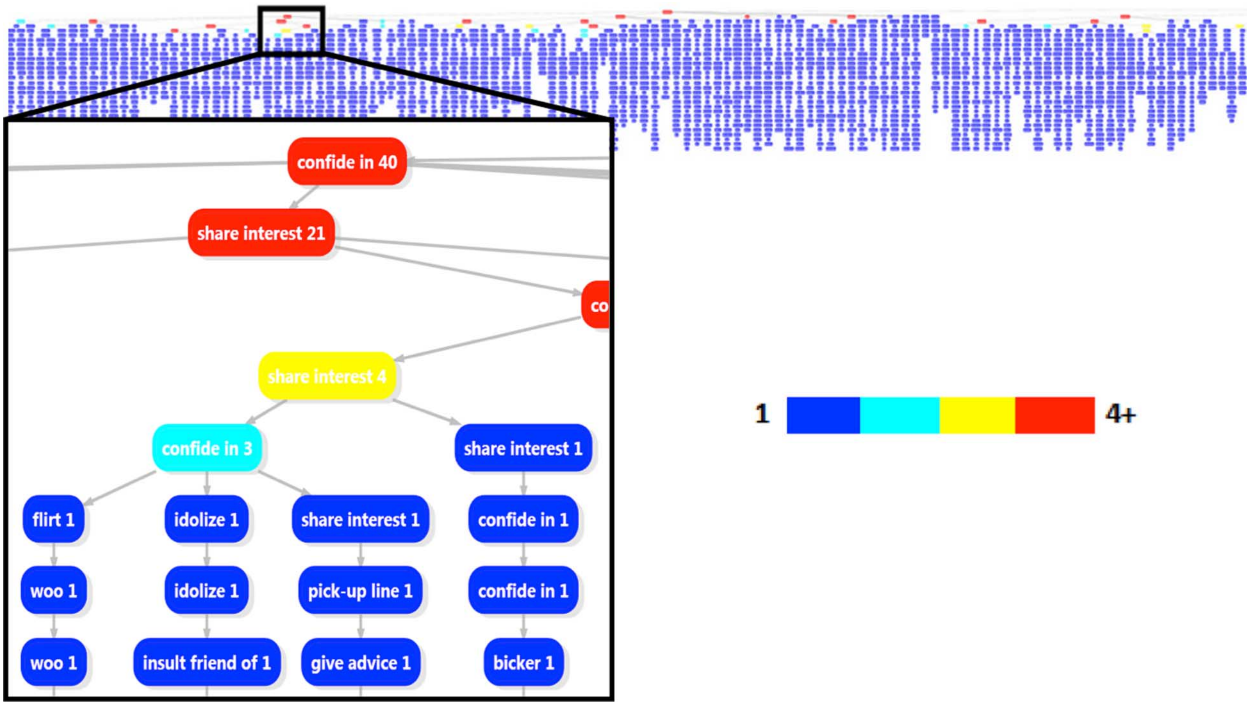

Fig. 3. Play trace graph showing how often each distinct path through Simon's story was traversed (shown by the number associated with each node, emphasized with color). The large band of nodes seen at the top of the diagram represents approximately one third of the total size of the complete graph. The cutout shows a section of the map in detail including examples of social exchanges (like "pickup line" and "confide in") that appeared in more than one play trace. The majority of play traces are unique.

Each play trace consists of the game events chosen by the player that have an effect on the social world, which are stored in the SFDB. The SFDB was designed to keep a record of CiF's activities, the social exchanges played, and enacted triggers rules. Additionally, *Prom Week* uses the SFDB to store when and how the player uses social influence points, a resource which players can use to push characters out of the comfort zone by making them behave out of character. When sent to the server, the SFDB is made into XML with included data about the level.

From when the game was officially released on February 14, 2012 to May 17, 2012, players have generated a total of 13 003 traces. Of these traces, 7074 took place in tutorial levels, 504 were of the goal-less freeplay mode, and the remaining 5425 took place in *Prom Week*'s stories. Only the 5425 story play traces generated in these first three months after the release of *Prom Week* are used in this evaluation.

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

2) Gameplay-Customized Story World 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 [40], [41], 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 desires of each individual player. Furthermore, story goals, level casts, and the desires of the characters themselves may restrict the options available in such a way that many players will be forced down a narrow few paths in their pursuit of story goals.

Before evaluating the variability of *Prom Week*, it is important to establish a baseline amount of variability. Some analysis of this baseline has been done on Quantic Dream's *Heavy Rain* [42], a game which places heavy emphasis on storytelling. The gameplay of *Heavy Rain* is split into small scenes, each having the player control one of four different protagonists in a certain location. Though every scene offers several opportunities for the player to make decisions, with the exception of the final scenes comprising the game's climax, these choices rarely have impact on the global narrative outside the scope of the scene they were presented in. Moreover, the variability presented within a scene often is either inconsequential (there are no consequences to choices made beyond an immediate response) or boils down to one of two cases: success or failure (either the player evades the cops or gets arrested). In short, most of the variability in the story game *Heavy Rain* is not meaningful.

In contrast, there is a very large degree of variation in the way that players navigated the social space of *Prom Week*. 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. Fig. 3 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 numeric values associated with nodes comprise a heat map indicating frequency of node visitation along that specific path; some are frequently visited (i.e., several players followed that exact same route up to the point of that node), while many are 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 branches sufficiently to demonstrate *Prom Week*, and CiF's, ability to provide 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 toward 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[1] revealed some interesting statistics on the patterns of sequences of social moves played. 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 ten times. With 6-gram analysis, there are 5066 unique patterns of social exchanges, one of which occurred 16 times, another ten times, and all the rest less than five 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.

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 also players are leveraging that variability. 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. 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 the user has seen and incorporates them into future social exchange selection. Moreover, their specific ordering also impacts the formulation of which social exchanges characters want to play with each other through the use of the SFDB and time-ordered rules, 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 move the average drops to 2.36. The average dips to 1.27 after the third turn, and hits 1.07 after the fourth. This illustrates the variability and impact that players have in their unfolding stories 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 a second goal of the system: 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 social influence points to unlock additional options).

The fact that, of these hundred starting options, only 11 were ever pursued between all of the gameplay traces implies that players are not choosing moves at random, but attempting to accomplish specific story goals. The beginning of each level provides framing text which contextualizes the characters' relationships to each other with respect 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; they are playing the story, not just experiencing it.

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 portions of the story paths—variable though they may be—need

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

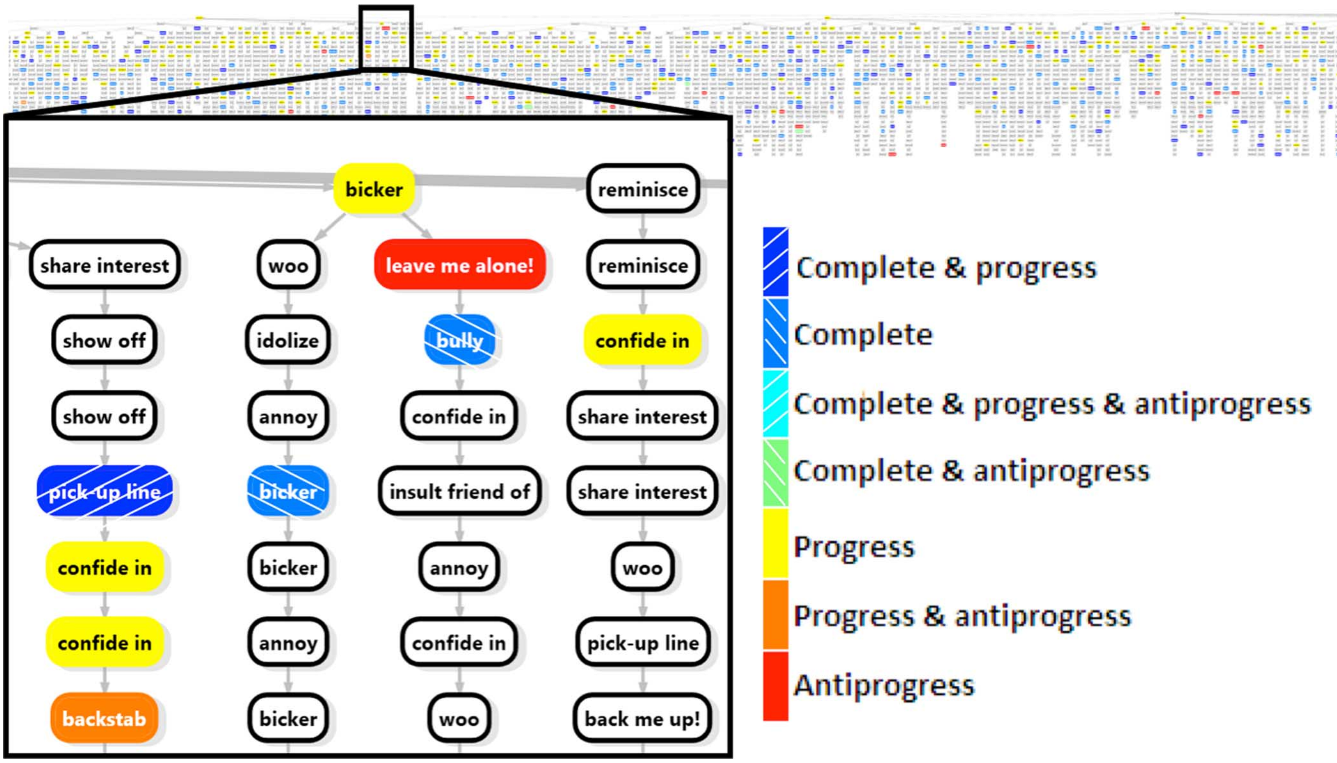

Fig. 4. Tree displaying the amount of progress toward goals in Simon's campaign. The color and texture of the nodes represents the type of goal progress. There are three types of goal progress that can be combined in any way. Complete means a goal was completed, progress means that one aspect of a goal was made true, and antiprogress 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 one-third of the total play traces of Simon's story.

to lead to successful goals. Story goals in *Prom Week* represent story states the player may make true in the story world. For example, in Simon's campaign, the player can choose to pursue 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 prewritten to leverage authorial control, there still exists template dialog 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.

4) Story Goal Completion: Fig. 4 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 progress toward the goals to moving the social state away from the story goal (antiprogress). These data were generated by taking the same level traces used to generate Fig. 3 and running them through CiF, keeping track of the goal accomplishments at each game turn.

Simon's campaign is the third nontutorial 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 Fig. 4, 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 toward 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 (an average of 2.32 goals per player). If every trace from every file had accomplished all five goals, the total would be 1315, which means that around 46% of all possible Simon goals were achieved. Goal progress was made a total of 837 times (an average of 3.18 times per player), and goal antiprogress was made a total of 44 times (an 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 games for most players, figuring out the nuances of the system to make story progress could have proven to be a challenge. And because goals are optional (and some are in direct opposition, impossible to accomplish together) we might expect few to be completed. The results are encouraging because not only were players motivated to pursue story goals, but also they were able to create a strong enough internal model of the storytelling system to be able to pursue story goals with some amount of success.

VI. Conclusion

In this paper, we have described CiF, an AI system enabling authorable models of social interaction between autonomous agents. While most previous interactive narratives have been tightly constrained to a few predetermined narrative options, usually offering the player high degrees of agency and freedom only in the context of combat or physics, not story, CiF enables highly dynamic and responsive social and narrative gameplay. The use of rules-based encoding of social norms, cultural knowledge, and appropriate behaviors for a given story world allows authors to explicitly encode social logic into a playable experience, creating rich and surprising character behavior. Retargeting social performances provides a route to making the authoring for such dynamic social spaces tractable. *Prom Week*, a fully realized game driven by CiF, has provided qualitative validation of this technique by receiving nominations from a number of major festivals for independent and experimental games, and the in-progress Mismanor and IMMERSE projects speak to the ongoing success of and interest in CiF's approach to dynamic social gameplay. We hope that CiF provides a strong proof of concept that building rules-based social worlds and retargeting social exchanges can become a successful approach for making game narratives more dynamic and responsive.

References

[1] J. H. Murray, *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. New York, NY, USA: Free Press, 1997, p. 324.
[2] Square Enix, *Final Fantasy XIII* 2010.
[3] Quantic Dream, *Heavy Rain* Sony Computer Entertainment, 2010.
[4] P. Harrigan and N. Wardrip-Fruin, Eds., *Second Person: Role-Playing and Story in Games and Playable Media*. Cambridge, MA, USA: MIT Press, 2007, p. 432.
[5] BioWare, *Star Wars: The Old Republic* Electronic Arts, Austin, TX, USA, 2011.
[6] Nintendo, Super Mario Galaxy 2 Tokyo, Japan, 2010.
[7] Valve Corporation, *Portal 2* 2011.
[8] J. McCoy, M. Treanor, B. Samuel, B. Tearse, M. Mateas, and N. Wardrip-Fruin, "Comme il Faut 2: A fully realized model for socially-oriented gameplay," in *Proc. Found. Digital Games Intell. Narrative Technol III Workshop*, 2010, DOI: 10.1145/1822309.1822319.
[9] R. McKee, *Story: Substance, Structure, Style and the Principles of Screenwriting*. New York, NY, USA: ReganBooks, 1997, p. 480.
[10] E. Goffman, *The Presentation of Self in Everyday Life*. Harpswell, ME, USA: Anchor, 1959.
[11] M. Lebowitz, "Creating characters in a story-telling universe," *Poetics*, vol. 13, no. 3, pp. 171–194, 1984.
[12] J. Meehan, "The metanovel: Writing stories by computer," Ph.D. dissertation, Dept. Comput. Sci., Yale Univ., New Haven, CT, USA, 1976.
[13] S. Turner, *The Creative Process: A Computer Model of Storytelling and Creativity*. New York, NY, USA: Psychology Press, 1994.
[14] B. Tearse, M. Mateas, and N. Wardrip-Fruin, "MINSTREL remixed," in *Proc. Intell. Narrative Technol. III Workshop*, 2010, pp. 1–7.
[15] C. Crawford, *Chris Crawford on Interactive Storytelling*. New York, NY, USA: New Riders Games, 2004, p. 384.
[16] J. Orkin and D. Roy, "The restaurant game: Learning social behavior and language from thousands of players online," *J. Game Develop.*, vol. 3, pp. 39–60, Dec. 2007.
[17] R. S. Aylett, S. Louchart, J. Dias, A. Paiva, and M. Vala, "Fearnot!: An experiment in emergent narrative," in *Intelligent Virtual Agents*, ser. Lecture Notes in Computer Science. Berlin, Germany: Springer-Verlag, 2005, vol. 3661, pp. 305–316.
[18] S. Marsella and J. Gratch, "EMA: A process model of appraisal dynamics," *Cogn. Syst. Res.*, vol. 10, no. 1, pp. 70–90, 2009.
[19] M. Si, S. Marsella, and D. V. Pynadath, "Modeling appraisal in theory of mind reasoning," *Autonom. Agent Multi-Agent Syst.*, vol. 20, no. 1, pp. 14–31, May 2009.
[20] A. S. Rao and M. P. Georgeff, "BDI agents: From theory to practice," in *Proc. 1st Int. Conf. Multi-Agent Syst.*, 1995, pp. 312–319.
[21] R. Evans, "The logical form of status-function declarations," *Etica Polit.*, vol. 11, no. 1, pp. 203–259, 2009.
[22] K. Erol, J. Hendler, and D. S. Nau, "Semantics for hierarchical task-network planning," TR 95–9, 1995.
[23] F. Charles, M. Lozano, and S. Mead, "Planning formalisms and authoring in interactive storytelling," [Online]. Available: www-scm.tees.ac.uk
[24] D. Isla, "Handling complexity in the Halo 2 AI," in *Proc. Game Develop. Conf.*, 2005, p. 12.
[25] Electronic Arts, *The Sims 3* The Sims Studio, 2009.
[26] R. Evans, "Re-expressing normative pragmatism in the medium of computation," in *Proc. Collect. Intentionality VI*, Berkeley, CA, USA, 2008.
[27] R. Evans, *Introducing PRAXIS: A Statically-Typed Logic-Programming Language for Modelling Social Practices*. Oxford, U.K.: Oxford Univ. Press, 2013.
[28] Linden Lab, *Versu* 2013.
[29] J. Porteous, F. Charles, and M. Cavazza, "NetworkING: Using character relationships for interactive narrative generation," in *Proc. Int. Conf. Autonom. Agents Multi-Agent Syst.*, 2013, pp. 595–602.
[30] J. I. Bakker, "The 'Semiotic Self': From Peirce and Mead to Wiley and Singer," *Amer. Sociol.*, vol. 42, no. 2–3, pp. 187–206-206, 2011.
[31] J. Davis, *Garfield and Friends* 1978 [Online]. Available: www.garfield.com
[32] S. Tak and H. Ko, "A physically-based motion retargeting filter," *ACM Trans. Graph.*, vol. 1, pp. 98–117, Jan. 2005.
[33] J. McCoy, "All the world's a stage: A playable model of social interaction inspired by dramaturgical analysis," Ph.D. dissertation, Dept. Comput. Sci., Univ. California Santa Cruz, Santa Cruz, CA, USA, 2012.
[34] A. Sullivan, A. Grow, T. Chirrick, M. Stokols, N. Wardrip-Fruin, and M. Mateas, "Extending CRPGs as an interactive storytelling form," in *Proc. 4th Int. Conf. Digit. Storytelling*, 2011, pp. 164–169.
[35] G. Yannakakis, J. Togelius, R. Khaled, A. Karpouzis, A. Paiva, and A. Vasalou, "SIREN: Towards adaptive serious games for teaching conflict resolution," in *Proc. 4th Eur. Conf. Games Based Learn.*, 2010, pp. 412–420.
[36] M. Lim, J. Dias, R. Aylett, and A. Paiva, "Improving adaptiveness in autonomous characters," in *Intelligent Virtual Agents*, ser. Lecture Notes in Computer Science. Berlin, Germany: Springer-Verlag, 2008, vol. 5208, pp. 348–355.
[37] C. Pearson, "Impressions: Prom Week," *Rock, Paper Shotgun*, 2012 [Online]. Available: http://www.rockpapershotgun.com/2012/02/16/impressions-prom-week/
[38] G. Costikyan, "Prom Week," *Play This Thing*, 2012 [Online]. Available: http://playthisthing.com/prom-week
[39] A. Stephens, "Prom Week," Story Wonk, 2012 [Online]. Available: http://alastairstephens.com/?p=87
[40] S. Sali and M. Mateas, "Using information visualization to understand interactive narrative: A case study on Façade," in *Interactive Storytelling*, ser. Lecture Notes in Computer Science. Berlin, Germany: Springer–Verlag, 2011, vol. 7069, pp. 284–289.
[41] S. Sali, "Playing with words: From intuition to evaluation of game dialogue interfaces," Ph.D. dissertation, Dept. Comput. Sci., Univ. California Santa Cruz, Santa Cruz, CA, USA, 2012.
[42] H. Wei and T. Calvert, "Conventions and innovations: Narrative structure and technique in heavy rain," Int. Digital Media Arts Assoc., 2013 [Online]. Available: http://idmaa.org/?post_type=journalarticle&p=376

Joshua McCoy received B.A. degrees in computer science and sociology/anthropology from Earlham College, Richmond, IN, USA, in 2004 and the M.S. and Ph.D. degrees in computer science from the University of California Santa Cruz, Santa Cruz, CA, USA, in 2010 and 2012, respectively.

He is a Postdoctoral Scholar and Lecturer at the Center for Games and Playable Media, University of California Santa Cruz. Through focusing on richly and tractably enabling social interactions between characters driven by AI, his research goals are to make interactive experiences featuring social interaction more prevalent and easier to create. Game design surrounding social play is a key component of this research. *Prom Week*, a social puzzle game featuring an AI system for social physics, is a game featuring the implementation of this research. He was the technical design lead for *Prom Week* and is currently working on Immerse, a simulation for training good stranger behavior via socially aware characters and fully bodied interaction funded by the Defense Advanced Research Projects Agency (DARPA).

Mike Treanor received the M.F.A. degree in digital art and the Ph.D. degree in computer science from the University of California Santa Cruz, Santa Cruz, CA, USA, in 2008 and 2014, respectively.

He is currently an Assistant Professor at the American University, Washington, DC, USA. He is a game developer and theorist whose research is aimed at finding new approaches for interpretation and expression of meaning within video games and computational media. Recently, he was a design and technical lead on Game-O-Matic, an expressive video game generator, and *Prom Week*, a social simulation game that was a finalist in both The Independent Game Festival and Indiecade. He has been an active participant in the field of game studies and has published on the subjects of video game interpretation, tools for game creation, social simulation, and procedural content generation.

Ben Samuel received B.A. degrees in computer science and in theater arts, the graduate certificate degree in theater arts, and the M.S. degree in computer science from the University of California Santa Cruz, Santa Cruz, CA, USA, in 2007, 2008, and 2012, respectively, where he is currently working toward the Ph.D. degree in computer science.

He has been a Teaching Assistant for courses on programming, acting, and interactive storytelling. He was the lead engineer on the *Prom Week* project. His research interests include the incorporation of theater theory into playable interactive narrative experiences, methods for creating socially aware digital characters, and systems which nurture procedural literacy.

Aaron A. Reed received the B.A. degree in film studies from the University of Utah, Salt Lake City, UT, USA, in 2005 and the M.F.A. degree in digital arts and new media from the University of California Santa Cruz, Santa Cruz, CA, USA, in 2011, where he is currently working toward the Ph.D. degree in computer science.

He has been an Instructor at the University of California Santa Cruz for a course on interactive fiction. He has previously worked on experimental game technology with the Microsoft Student University Program, and was the lead writer on the *Prom Week* project. His research interests include the intersection of computation with written and oral storytelling, author assist tools for interactive narrative, and emergent behavior in dynamic storytelling systems.

Michael Mateas received the B.S. degree in engineering physics from the University of the Pacific, Stockton, CA, USA, in 1989, the M.S. degree in computer science from Portland State University, Portland, OR, USA, in 1993, and the Ph.D. degree in computer science from Carnegie Mellon University, Pittsburgh, PA, USA, in 2003.

He is a Professor of Computer Science, Co-Director of the Expressive Intelligence Studio, and Director of the Center for Games and Playable Media at the University of California Santa Cruz, Santa Cruz, CA, USA. His research focuses on expressive intelligence, which explores the intersection of artificial intelligence, art, and design. Along with A. Stern, he created Façade, which is the world's first fully produced, real-time, interactive story. Recent projects include *Prom Week* (2012 finalist at IndieCade and the Independent Games Festival) and Immerse [a Defense Advanced Research Projects Agency (DARPA)-funded project on teaching good stranger behavior through immersive full-body interaction with socially aware virtual characters].

Noah Wardrip-Fruin received the M.F.A. degree in literary arts and the Special Graduate Study Ph.D. degree from Brown University, Providence, RI, USA, in 2003 and 2006, respectively.

He is an Associate Professor of Computer Science, Director of the Digital Art and New Media MFA program, and Co-Director of the Expressive Intelligence Studio at the University of California Santa Cruz, Santa Cruz, CA, USA. He is the author of two books: *The New Media Reader* (Cambridge, MA, USA: MIT Press, 2003) and *Expressive Processing* (Cambridge, MA, USA: MIT Press, 2009). His recent projects include *Prom Week* (2012 finalist at IndieCade and the Independent Games Festival) and the 2012 Media Systems gathering [with partners the National Endowment for the Humanities (NEH), the National Endowment for the Arts (NEA), the National Science Foundation (NSF), Microsoft Studios, and Microsoft Research].