

## ISRST: GENERATING INTERESTING MULTIMEDIA STORIES ON THE WEB

**Arturo Nakasone<sup>1</sup>, Helmut Prendinger<sup>1</sup>, and Mitsuru Ishizuka<sup>2</sup>**

<sup>1</sup>*Digital Content and Media Sciences Research Division,  
National Institute of Informatics, Tokyo, Japan*

<sup>2</sup>*Department of Information and Communication Engineering, Graduate School  
of Information Science and Technology, University of Tokyo, Tokyo, Japan*

□ *Interactive storytelling (IS) models, either implicitly or explicitly, have had to deal with three major aspects in their implementations, regardless of their technology: (1) how to define story events, i.e., the pieces of information that constitutes the content of any story; (2) how to present story events, i.e., the algorithm that is used to convey this content to the user; and (3) how the user is to interact with the story, i.e., the variables that the user can modify in order to change the presentation of the events in content or in order. The majority of IS implementations encode in their definition of story event timing information (the range of time specified in terms of the story timeline in which the event must be presented) and interaction variables associated with that predefined timing. This approach is convenient for story presentation, but presents several restrictions in terms of story understanding and story dynamism. To overcome these restrictions, we developed interactive storytelling model using rhetorical structure theory (ISRST), our proposal for a storytelling model based on the organization of generally defined events using a subset of rhetorical relations proposed by the RST and the application of narrative principles and user interaction through interest to generate appealing stories. This presentation of ISRST will be complemented by a critical discussion about technical and usability issues of its implementation. We also conducted an empirical study using a real full-fledged story, which suggests that a user's story satisfaction can be associated with the presentation of a story outcome that emphasizes the user's empathetically preferred character(s).*

The authors are indebted to Prof. Federico Peinado from Universidad Complutense de Madrid for his important contribution regarding the graphical aspects of the test story, and to Prof. José Antonio Pow Sang from Pontificia Universidad Católica del Perú for his valuable support during the testing of ISRST.

Address correspondence to Arturo Nakasone, Digital Content and Media Sciences Research Division, National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda Ku, Tokyo 101-8430, Japan. E-mail: arturo.nakasone@gmail.com

## INTRODUCTION

In recent years, researchers in the fields of education and entertainment have started to give greater importance to the implementation of concepts related to storytelling and narrative in their work. Both the enhancement of the knowledge acquisition process in learning and the satisfaction obtained in entertainment through multimedia applications are much better conceived if we take into consideration that human experience about acquisition and understanding of its reality is framed into the context of telling and receiving information through stories. Even though storytelling applications were oriented to text generation at first (e.g., Elliott, Brzezinski, Sheth, and Salvatoriello 1998), their range of applicability increased due to the ubiquitous presence of communication networks such as the Internet and the prompt availability of multimedia content. In this context, we can find applications that have diverse objectives which go from document generation (Callaway and Lester 2002) to virtual reality presentations and multimodal dramatizations (Mott and Lester 2006). The increase in technology made possible not only the creation of complex stories with outstanding content, but also a more enhanced interaction in which the user can completely personalize, through preferences or actions, his/her experience in terms of “what to see” (the story content) and “how to see it” (the story presentation).

Current approaches to interactive storytelling encode in their story content the timing information, which identifies the range of time in terms of story timeline in which each piece of story content must be presented, and the variables that the user is allowed to modify in order to interact with the story. This approach, although convenient for story presentation, presents several restrictions in terms of story understanding (e.g., if the content is not adequately organized) and story dynamism (e.g., algorithms for story presentation have almost no flexibility to arrange the story content based on its narrative appeal to the user).

To overcome these restrictions imposed by the inclusion of the timing information in the content of a story, a storytelling system should eliminate this need by implementing a timeless way to link story content and providing an interaction scheme that is independent of such content. Therefore, we developed storytelling model using rhetorical structure theory (SRST) (Nakasone and Ishizuka 2006), a storytelling application that presented stories using a story content organization based on the tree structure proposed by the RST (Mann and Thompson 1987). Although SRST implemented the processing of explicit semantic links between events, this application lacked a coherent way to include an unobtrusive user interaction due to the constraints for event sequencing of the RST tree itself. In order to solve this problem, we present ISRST, our proposal for a storytelling model based on the organization of generally defined

events using only a meaningful subset of rhetorical relations proposed by RST and narrative principles combined with interest measurement to provide a unique interactive storytelling experience. This work is presented as an extension of the system described in Nakasone and Ishizuka (2007).

The remainder of the article is organized as follows: the next section covers the description of related research projects, classifying them based on the way they define, present, and let users interact with their story content. Then, we point out the common characteristics of the previously described applications from the model point of view and present the differences with the model proposed by our approach. In the following section, we discuss the current version of the ISRST model in detail. Next, we present the technical aspects of our web implementation for ISRST, along with a section detailing the results and conclusions of our performed study using a real full-fledged story called “When Your Heart Takes Over,” which is currently available for testing at ISRST (2009). Finally, a section that presents a critical discussion about technical and usability issues regarding the implementation of ISRST and our plans for future research conclude the article.

## **RELATED WORK**

In this section, we describe the current state of the art in storytelling by classifying applications into three main categories: story events (i.e., how the story content is defined), story engine (i.e., how the story content is presented), and story interaction paradigm (i.e., how users interact with the story content). The purpose of this section is to provide a general overview of the storytelling models, indicating particular strengths and limitations for applications in each of the different approaches to story engine implementation. In the following subsection, we identify and describe in detail the common limiting factors for these storytelling models, and how our approach deals with those factors.

Despite the wide range of storytelling models and implementations, the core issue around which any storytelling application revolves is the fact that stories are “sequences of events, mental states, or happenings” (Bruner 1991). This simple but accurate definition stresses the fact that: (a) stories are composed of entities called “events” (this definition encompasses the concepts of mental states and happenings in ISRST), and (b) these story “events” are conveyed as a sequence to form a story. Taking into consideration these basic attributes for any story, storytelling applications have to—either implicitly or explicitly—deal with the following concepts in their implementations, regardless of their objective (e.g., information presentation or dramatic storytelling).

## Story Events

In a general sense, story events are defined as the smallest units in which information or situations are presented and are meaningful to the advancement of the story as a whole. The concept of “smallest unit” does not refer to the fact that an event should have only one component, but that regardless of the amount of components, an event must represent a unit in terms of its situational meaning in the context of the whole situational aspect of the story. In dramatic environments, this is reflected in the enforcement of the Aristotelian principles of unity in time (i.e., an event takes place in a limited and sequential time-frame), place (i.e., an event’s scenario is constrained to one location), and action (i.e., an event must not contain actions that are extraneous to the general plot of the story) (Sheldon 2004). In presentation environments, the concept of unity in events is slightly different (e.g., an event must present only one topic and be constrained to a particular context), but it refers to the same characteristics nonetheless. In the majority of storytelling applications, events are represented as plain text (Figa and Tarau 2003; Silva, Guilherme, and Paiva 2003; Sumi and Tanaka 2005), scripted individual character actions (Gebhard, Kipp, Klesen, and Rist 2003), story world states (Magerko and Laird 2003, 2004; Szilas 2003), or multimedia content (Callaway et al. 2002; Rober, Huber, Hartmann, Feustel, and Masuch 2006). In ISRST, we support multimedia content in the form of images, audio, and video distributed over the web.

## Story Engine

The story engine implements the algorithm that is in charge of sequencing events to form a story and constitutes the core component in any storytelling application. Researchers in the storytelling field have developed very creative techniques that make use of well-known algorithms based on artificial intelligence (AI). According to these techniques, storytelling applications can be classified into the following categories.

### *Rule-Based Applications*

The story engine makes use of elaborated logical rules in the form of antecedent—consequent pairs in order to select the next event based on what the user performed or selected in his or her previous interactions with the application. Some rule-based applications acknowledge that certain dramatic effects must exist in the story; therefore, they normally categorize their rules according to a relatively fixed narrative pattern. This narrative pattern may proactively involve the user as a protagonist, as in a dialog exchange (Figa and Tarau 2003), or make the user take

meaningful decisions in terms of narrative through the manipulation of the story protagonists and/or the story environment, as in the case of the Erasmatron application (Crawford 2005). Other rule-based applications take into consideration not only a narrative effect on isolated rules, but also on the whole set. This narrative effect takes the form of functions that help to shape the story dynamically by ranking the rules based on its narrative impact (Szilas 2003) or by analyzing a story progress based on its dramatic “tension” and simulating possible future situations in terms of user interaction (Mateas and Stern 2003). The utilization of rules to define the course of a story is one of the most natural ways to create event sequencing, and results achieved with these kinds of applications in terms of story construction are encouraging. However, the creation of event content can take more time than usual for a full fledged story, since the processing of the user interaction is heavily dependent on the situation portrayed in each event. Moreover, as the story increases in complexity and duration, it can become difficult to maintain the knowledge of antecedents and consequents for every event in the story. Our approach eliminates the need for direct antecedent—consequent knowledge maintenance, since event sequencing in ISRST is done through semantic relations.

### ***Goal-Based Applications***

A goal event or events are established as the final outcome of the story. From a set of initial conditions, a story is unfolded by the sequence of events that are needed to reach such goal event(s). Planners and their variations are commonly used in these applications and, therefore, events have to be annotated with a set of pre and/or post conditions to satisfy the planners’ requirements. One of the most widely used planners is the hierarchical task network (HTN) planner, whose structure is conveniently used to organize story events based mainly on their content. In some of these applications, the user’s actions are not controlled by the story engine at all times, which might affect the narrative quality of the story during the planning process. This problem has been solved, for instance, by classifying the actions of the user in order to know beforehand which kinds of actions may be disruptive (Riedl et al. 2003). By taking this HTN structure to its limits, some researchers have developed structurally very complex stories regarding action selection capabilities in the form of optional events (Cavazza, Charles, and Mead 2002). By fixing the event sequencing algorithm, goal-based applications help reduce the burden of creating stories, allow authors to mainly concentrate on the story content itself. One of the major disadvantages for these kinds of applications is that events are constrained by the set of preconditions and/or post conditions that are attached to them. In other words, they occupy a rather fixed position in the planner’s content structure, leaving little space for flexibility during

the story construction process. Interactive storytelling model using RST eliminates the need for preconditions and/or post conditions attached to events.

### ***State Transition-Based Applications***

They define events as state nodes that specify the current contextual situation in a particular point of the story. Bayesian networks, finite state machines, and their variations are commonly used. Therefore, events must be annotated to fit the node requirements in the network or machine. State nodes may also include the content of the story itself (i.e., the diegetic and/or mimetic components of a story), whose complexity goes from simple pregenerated scripts for agent manipulation (Gebhard, Kipp, Klesen, and Rist 2003) to free motion and interactive experiences using virtual reality technology (Swartout et al. 2001). In order to enforce narrative quality into these kinds of applications, several external functions are embedded into the network or machine in order to regulate the selection of nodes. In some cases, the nodes that represent story events are separated into levels to reflect the different stages in the telling of the story (Sobral et al. 2003). More complex schemas allow the inclusion of time transition constraint factors to minimize the disruption in the flow of the story caused by the user's actions. These systems use a predictive model of the user's actions in order to maintain the coherence in the plot (Magerko and Laird 2004). State transition-based applications, like their goal-based counterparts, fix their event sequencing algorithms to facilitate story transition implementation to content authors. However, they offer little story control inside each node, which could degrade narrative quality if the content of the node becomes too large. This problem could be overcome by defining shorter events and a greater number of nodes, but this might increase the complexity of the state transition conditions. Although we face the same kinds of difficulties in ISRST regarding the size of events, we provide a more flexible sequencing algorithm for richer storytelling experience.

### ***Template-Based Applications***

Events are selected to fit story templates and permuted to create new narrative experiences based on those templates. Since the story template constrains which and how events are presented, these events must be annotated to define their role in the context of the whole story. In some applications, templates are implemented using AI techniques such as case-based reasoning (CBR) (Gervas, Diaz-Agudo, Peinado, and Hervas 2005). Templates are normally used by applications that deal with a specific topic (e.g., a bullying situation (Sobral, Machado, and Paiva 2003), which limits their use in other contexts. Template-based applications provide a

convenient way to generate stories with a high content diversity based on story patterns that are fixed in advance. However, given the rigidity of the template itself, user interaction is normally constrained to the specification of the story plot, and/or to the preprogrammed situations created for each event. Our approach provides more flexibility in terms of storytelling through the implementation of a narrative quality evaluation module.

### ***Script-Based Applications***

These applications are a particular case of template-based applications in which stories are described using a high level language to create story scripts. The application, then, presents the events in the way specified by such scripts. Some scripts languages are based on common standards. For instance, in some cases, sequencing specification languages such as the synchronized multimedia integration language (SMIL) are used to define event sequencing processes for presentational purposes (André, Concepcion, Mani, and van Guilder 2005), and in other cases, content specification languages like the global document annotation (GDA) are used to organize the content that later would be used to generate animated stories (Sumi and Tanaka 2005). Other script languages have more proprietary characters, and their purposes vary from story flow control to definitions of characters and interrelationships (Sgouros, Papakonstantinou, and Tsanakas 1997). Script-based applications offer more flexibility to story control in terms of event sequencing than the template-based ones, but they still lack flexible user interaction methods. They are well suited for presentation types of stories, in which the content can be automatically retrieved from databases or the Internet, and, unlike template-based applications, the manipulation of the story's event-sequencing process through user interaction can be specified in the story script, although just to a small degree. In contrast, ISRST implements a more flexible and unobtrusive way that allows users to interact with a story through their interest specification at any moment.

### **Story Interaction Paradigm**

The story interaction paradigm defines the way in which the user is allowed to interact with the storytelling application by changing the conditions inside the engine in order to personalize his or her story experience. Even though there are some applications that do not offer any interaction to the user (André et al. 2005; Geurts, Bocconi, van Ossenbruggen, and Hardman 2003; Nakasone and Ishizuka 2006), a lot of them do provide varied types of mechanisms for interaction, which goes from parameter specification (Callaway et al. 2002; Gervas et al. 2005) and menu selection and interruptions (Crawford 2005; Little, Geurts, and

Hunter 2002; Sumi and Tanaka 2005; Szilas 2003) to full user action multimedia processing (Cavazza et al. 2002; Magerko and Laird 2003; Mateas and Stern 2003; Riedl et al. 2003; Swartout et al. 2001). In all these cases, the interaction paradigm reflects the technology with which the application itself was implemented (e.g., menu selection in text processing, game control manipulation in video gaming, parameter configuration in slide presentations). Another common attribute of all these types of interaction paradigms is that they define a somewhat obtrusive way of manipulating users' preferences during the presentation of a story, which might hinder those users' experience in terms of satisfaction. In ISRST, we implemented a more unobtrusive alternative through the processing of user interest (see the "Attributes for a Real Semantically Conditioned Solution" subsection for more detail on the advantages of our approach).

### **Critical Assessment of Current Storytelling Models**

One common characteristic of all the previously referred applications is that they include in their story events an attribute called timing information, which is the range of time in the context of the story timeline in which an event must be shown. Since the range of time in which an event must be presented is also fixed due to the role it plays in the story in terms of content, events also have an implicit time-frame in which they are to be presented even if their timing information is not explicitly encoded. For instance, in the case of the bullying story (Sobral et al. 2003), an event telling about the fight between two characters (Event A) must be presented before the event that tells about the resolution of such a fight (Event B). Even though the timing information defines a relation of precedence between events A and B, it does not necessarily mean that the former must *immediately* precede the later. Since the timing information was not defined as a point in time but as a range of time, storytelling applications have had some flexibility to organize their story events as long as their timing information was respected (e.g., applications organize their story events into databases, as document files, as nodes in state machines, as tasks in hierarchical networks, etc.).

Another characteristic is that story events also include in an explicit way the interaction variables the user manipulates to shape the story. These interaction variables can be defined in the event as motion processing parameters, content selection alternatives, voice commands, etc. Since the interaction variables can differ from event to event as the story progresses, each event has a unique set of conditions that determine which variables are available to the user and which ones are not for that particular event. For instance, at one point in the story, a character may be allowed to utter commands, but at another point, he/she may only



have the option to move to different places throughout the scene (Riedl, Saretto, and Young 2003). The story engine saves the modifications to these interaction variables either as changes to the story environment (story profile) or as changes to the preferences of the user (user profile). Once the changes have been processed in the profile, the story engine uses this profile to select the next event in the story. We refer to these types of storytelling applications as time-conditioned applications (see Figure 1a).

Time-conditioned applications are common since both the construction and the telling of a story are devised into a framework based on timed events and sequences which both authors and users are comfortable with and understand thoroughly. If stories are constructed in this way, the user's understanding during their presentation might be affected if the timing information in the events is not adequately set. In other words, if the story is not well constructed from the beginning, the user may obtain a less than satisfactory experience from it. Moreover, since the timing information constrains when an event can be presented, it may also restrict the versatility of the story engine to adapt to the user's unsatisfactory experience in real-time. In order to overcome this problem, some researchers have explored other approaches to create stories, not based on events with a preconceived timed organization, but on events organized using meaningful time-independent relations.

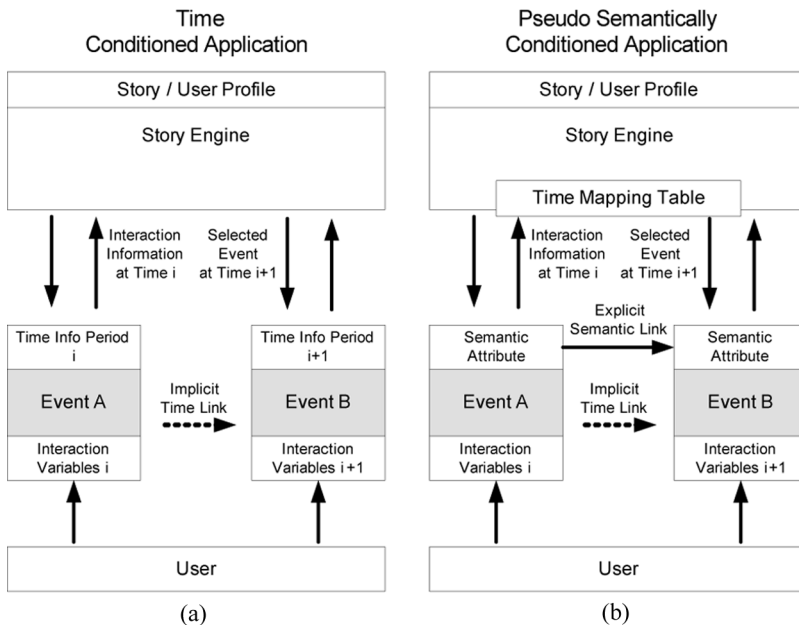


FIGURE 1 Current storytelling design paradigms.

Semantic relations have been considered a good alternative since the knowledge acquired by human beings is obtained through the assimilation of events that form in their minds, not only a time-related organization, but an interconnected set of situations linked by relations that bear some meaning. However, not all semantic relations can be used to interconnect story events. For instance, some semantic relations define nontemporal links between static objects (e.g., A is the father of B, C is the college in which B studied, etc.), but these kinds of relations are not suitable for story events since static objects themselves do not conform to the definition of event in terms of temporality (i.e., an event is to tell about a situation or present a fact that develops through time). Relations that are rhetorical in nature (e.g., A is the cause of B, C describes the context of D, etc.) are much better suited to address the time-changing nature of the events. Temporality, in this case, addresses the concern of time inside an event, not as links between events.

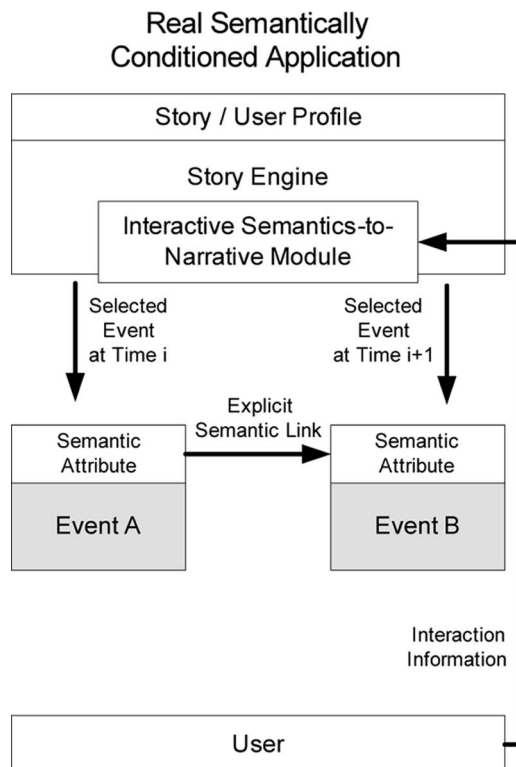
Semantically organized events have been used in applications that present multimedia content using primarily limited subsets of the relations defined by the RST proposed by Mann and Thompson (1987). For instance, some applications have used rhetorical relations to organize their content based on its discourse properties (Callaway *et al.* 2002), and others in more complex organizations such as complete presentational ontologies (Geurts *et al.* 2003). The use of rhetorical relations in narrative generation (i.e., event sequencing) is very limited in comparison to the use of the same relations in event organization. In most cases, the story event sequencing process is hard coded into a set of fixed rules, which map rhetorical relations to a set of spatio-temporal relations, that is, relations concerning space and time, which are more akin to the temporal event sequencing observed in storytelling applications (Little *et al.* 2002). We refer to these types of storytelling applications as pseudo semantically conditioned applications (see Figure 1b). Even though these kinds of applications eliminate the use of timing information attributes, the interaction variables are still encoded in the definition of events, generating an implicit time relation and defeating the purpose of the explicit semantic relation.

### **Attributes of a Real Semantically Conditioned Solution**

Therefore, a real semantically conditioned application would require that the interaction variables defined in the story events be transferred to the story engine in order to eliminate the generated implicit temporal relation. Since interaction variables are strongly dependent on the technology used to create events, some of their capabilities would be lost in the process. Thus, the interaction scheme in the story engine must be generic enough to be applied to diverse types of event creation technology

and flexible enough to allow users to specify their preferences without compromising their experience in the story.

Current interaction paradigms, which are based on parameter specification, menu selection, or multimedia action processing, are not only dependent on technology but also have very marked obtrusive characteristics (i.e., users are directly asked about their content preferences). Even though they constitute accurate representations of the interests of the users, in some cases their level of obtrusiveness may hinder the user experience in terms of the natural flow in the story (Csikszentmihalyi and Csikszentmihalyi 1988) and accentuate the dependency between the interaction paradigm and the story events themselves. In general, it can be stated that, regardless of the kind of interaction the user experiences, interaction variables are used to reflect the user's interest in the story as a whole or in a particular portion of it. Hence, interest is perceived as the most important emotional attribute when interacting with a story (Tan 1996); and a truly independent interaction scheme must be able to reflect this interest value into the event selection process at any time during the presentation of the story,



**FIGURE 2** ISRST storytelling design paradigm – real semantically conditioned application.

and not only at specific points. By making this interaction as unobtrusive as possible, it can be assured that the user's experience in the story will be maximized while maintaining the smoothness of the story flow itself (see Figure 2).

In conclusion, a real semantically conditioned application would have to implement the following aspects:

- Event annotation with the exclusive use of semantic relations and the implementation of a framework for story authoring using these concepts.
- A story engine that implements a semantics-to-narrative approach to event sequencing (i.e., how to generate a sequential story based on events' relations).
- An interaction paradigm based on generic real-time interest specifications.

### **THE ISRST MODEL: INTEREST INTERACTION ON A SEMANTICALLY RELATED EVENT FRAMEWORK**

In ISRST, our approach to implement a real semantically conditioned application was based on the following issues:

- The specification of the conceptual ideas that are used to define what a story event is and how semantic relations are to be described.
- The creation of an ontological framework, which encompasses not only the necessary elements to define story events and their semantic relations, but also the necessary support structure to deal with the proper authoring of stories. In ISRST, we implemented this ontological framework using the web ontology language specification (OWL).
- The principles by which narrative properties are included into the story engine, establishing how semantically related events are conveyed into a linear narrative sequence.
- The implementation of a user interaction paradigm based on interest, and how this interest is used to shape the story experience.

#### **Conceptual Ideas Used in the ISRST Model**

The conceptual ideas implemented in the ISRST model cover four aspects: story content and organization, story thematic, story characters, and story variability. For story content and organization, we addressed the problems of content diversification in a story event (i.e., different types of content in an event) and appropriateness of rhetorical relations for event interconnection. Regarding story thematic, we analyze how story topics or themes are specified in a story. The story characters aspect addresses the issue of agency in events and how it is defined in the semantic framework.

Finally, story variability refers to the different possibilities in terms of story content that the model can handle.

### *Story Event*

A story event is defined as the minimum unit by which a story or part of it can be told. Any event defined in our model corresponds to an object and is uniquely identified in order to comply with the particularity property of narratives (Bruner 1991), which states that events must be composed of happenings that are unique and not repeated in other events. Since the content of each event is expected to vary from application to application, our event object has no internal structure. Rather, it is left to the criteria of the author of the story, the organization of the content inside the object, to achieve maximum flexibility for his/her intended storytelling implementation. Figure 3 shows an example of an event definition for our test story in OWL as well as descriptions for its main components.

In our web implementation, we defined events by simply making a reference to an external text file in which the actual event content is stored. This definition allows us to have a less cluttered and more orderly OWL representation of the story. The content of the event is specified through a proprietary language called event specification language (ESL). This language contains an appropriate set of instructions for multimedia content manipulation (text, audio, and video file processing) and synchronization. The details concerning ESL and event file content will be explained in “The Presentation Module” subsection.

### *Rhetorical Relations*

It can be argued that stories in our minds are much more than a linear temporal sequence of events (e.g., Graesser and Clark 1985). In fact, stories have been characterized as “complete patterns that communicate a special kind of knowledge to our pattern recognizing mental module” (Crawford 2005). Since these patterns are present in our minds as a web

```

                                Name of the Event
<owlx: Individual owl: name="#BREAKING_DOWN">
  <owlx: type owl: name="#Event"/>
  <owlx: DataPropertyValue owl: property="#hasAction">
    <owlx: DataValue owl: datatype="xsd:string">
      FILE@@http://localhost/arturo/isrst/events/Breaking_Down.txt
    </owlx: DataValue>
                                Content of the Event specified in a file
  </owlx: DataPropertyValue>
</owlx: Individual>

```

FIGURE 3 Example of event definition in OWL.

of interconnected events, it is clear that such connections deal with much more complex relations than the ones constrained by temporal linking.

Natural text is considered one of the oldest ways to transmit a story and has been extensively analyzed in order to discover which rules govern its generation not only in terms of text organization, but also in terms of narrative characteristics. From a semantic point of view, rhetorical theories have provided us useful ways to analyze which kinds of relations can be defined between blocks of text in a narrative. The RST (Mann and Thompson 1987) is the most accepted theory, since it was created as a tool for computer text generation and its study has sparked the development of a full linguistic theory on its own. Rhetorical structure theory was intended to deal with the description of text rather than its creation or understanding. Given a certain piece of text, an RST analysis of such text gives as a result a structure that identifies the minimum units in the text in terms of semantic meaning (events) and the rhetorical relations that link them. Figure 4 shows a simple text sample and its corresponding RST analysis.

Given this analysis, we could infer that rhetorical relations are useful to describe semantic relations not only between blocks of text, but also between ideas, concepts, and events in a broader sense. Relations such as CAUSE (i.e., one event is the cause of another) or BACKGROUND (i.e., one event serves as background information for the other) that can be inferred from a text reflect not only the organization of those pieces of information in the text, but also the meaning that those events had in the mental story pattern of the author of such text. Even though the way

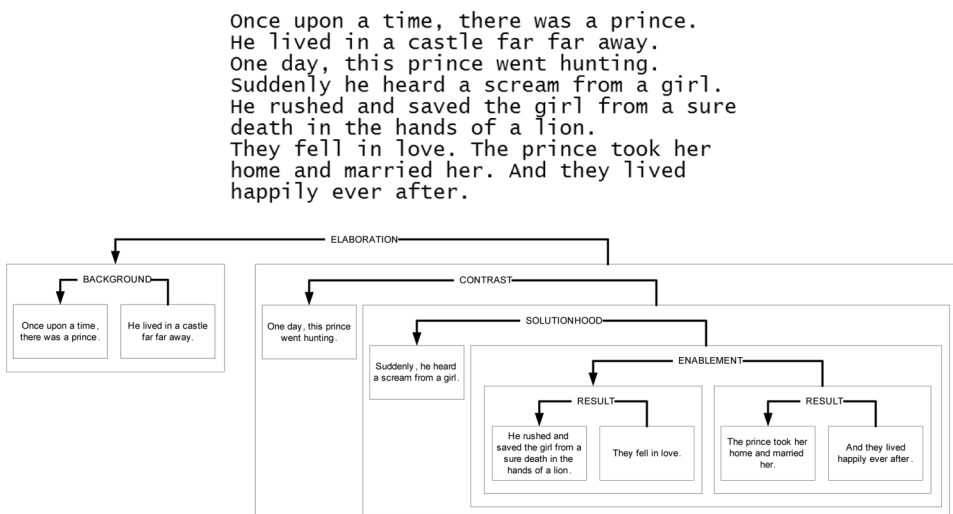


FIGURE 4 RST text sample analysis.

by which events are linked in a story are, in general, consistent with the definition of the complete set of RST relations, we believe that the use of this set turns out to be impractical for the purpose of event content creation, and people might find it easier to use a much smaller set of relations when they construct and remember their stories.

Most narratologists agree that the most important rhetorical relation between events is the causal one (e.g., Bruner 1991). Nevertheless, we may argue that this is not the only relation that is taken into consideration when human beings create story patterns in their minds. As discussed by Tan (1996), when we observe sequences of actions in a film, “the cognitive concern corresponds broadly to the assimilation of the action observed into a canonic narrative structure. This concern is also satisfied by a wealth of different relationships other than causal ones within the ultimate cognitive representation of the narrative.” Given the fact that a set of different types of relations help assimilate the content of a story, it is plausible to assume that the same set of relations may also be used to create such content. Therefore, we selected an appropriate set of rhetorical relations taking into consideration the flexibility and unambiguous character of their semantic definitions.

For the current version of ISRST, the following relations were considered:

- *Background*: In this relation, one event A is referred to as the context in which another event B happens. Event A may be located in the past of event B, but it does not necessarily entice a cause and effect relation.
- *Cause*: In this relation, one event is identified as the cause of another. The cause relation is one of the main relations in the ontology because it is the relation between events that people clearly distinguish.
- *Purpose*: This relation reflects the necessity for one event to be shown before another can be shown. Even though it is similar to the cause relation, its use implies a condition for advancing in the story, discussed in the “Qualities of Narrative in ISRST” section.
- *Result*: This relation indicates that an event is shown as a direct consequence of another event. It is also linked to the purpose relation, but has more immediate and final connotations. This relation is mostly used to indicate the presentation of final events in part of the story or the story itself.
- *Contrast*: For any story to have narrative quality, it must show some kind of conflict between two or more events (Bordwell 1986). Conflicts, implemented as contrasts, give stories the opportunity to enhance their audiences’ interest by creating narrative tension.
- *Solutionhood*: This relation provides a way to define how a contrast relation will be solved.

- *Elaboration*: In this relation, an event is shown to give more details about another event.
- *Evaluation*: This relation provides a way to state a final conclusion about one event. It is helpful for authors to convey a final message or thought to the story.
- *Sequence*: This relation establishes a linear temporal link between two events. This relation is useful to enforce sequencing in stories, but it should not be used as the primary way to link events.

### Story Concepts

Stories are always developed around a particular set of concepts. Concepts are important because they help regulate expectations with respect to the course of action in the story (Tan 1996) and, therefore, are useful to define the internal boundaries of a story regarding its content. These internal boundaries, normally conceived in the form of scenes or episodes, are understood to be the smallest units for concept association. Concepts themselves are not isolated, since the internal boundaries that make up a story are related to one another, not only by semantic relations, but also by a transition in concepts that determines the smoothness of the story in terms of knowledge presentation. In order to implement this transition in ISRST, we have created a graph-like structure of directed linked concepts known as the story concept network. As an example, Figure 5 shows the logical design schema of a story concept network which defines several topics that may constitute a common day for a person.

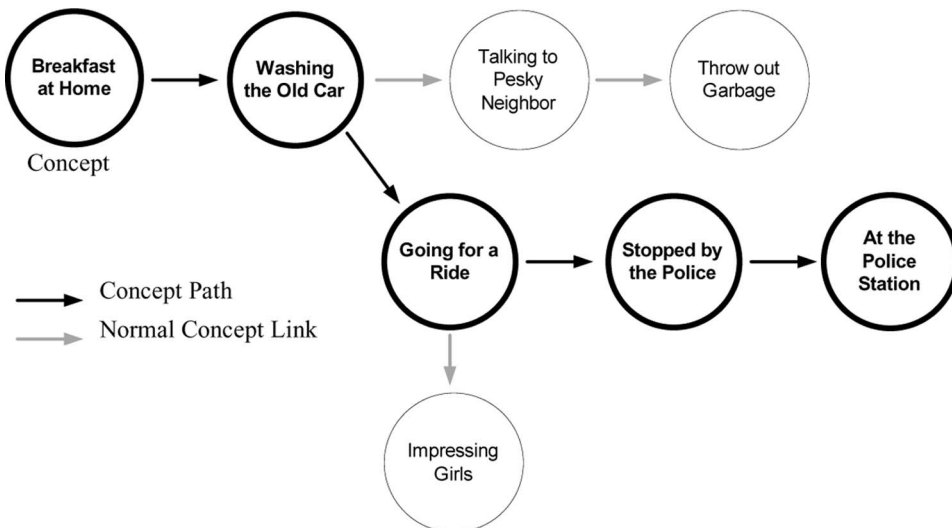


FIGURE 5 Logical design schema of a story concept network.



In a story concept network, each node represents a concept and each link represents a directed relation that defines the dependency in terms of knowledge precondition between the two connected concepts. For instance, since the “Breakfast at Home” concept is directly connected to the “Washing the Old Car” concept, it is said that “Breakfast at Home” must come before “Washing the Old Car” in the story for it to be coherent in terms of knowledge presentation. In the case of two concept nodes that are not directly connected, a logical transition in the network between these two nodes must exist in order to guarantee such coherence. This transition is known as the concept path. If we want to generate a story in which concepts “Breakfast at Home” and “At the Police Station” must be present (i.e., a story that depicts why this person ended up in a police station), the concept path formed by the intermediate nodes “Washing the Old Car,” “Going for a Ride,” and “Stopped by the Police” must also be included to ensure a smooth flow of knowledge for the whole story.

The main advantage for the implementation of this schema lies in the fact that it allows a flexible generation of stories based on concept preselection (i.e., a story can be defined as a collection of key concepts and the concept path is automatically generated) and on-the-fly concept rearrangement (i.e., a story can modify its set of concepts in real-time, provoking changes in the concept path in real-time also). Even though directed links are the simplest way to link concepts, the story concept network accurately represents the dependency and precedence relations that are to exist between knowledge-based nodes.

#### ***Agents and their Role in Semantic Relations***

Stories are about characters and these normally act as target objects for the attention of the user, since understanding them can produce in him/her not only pleasure in terms of knowledge acquisition, but also expectations of huge gains in satisfaction due to his/her empathy towards them. In ISRST, we define these characters as agents. Like the story event definition, an agent definition refers to a general concept regarding an entity into which a user pours his/her attention and, therefore, has vested interest in.

In the general definition of story, events are constituted by agents that perform or acknowledge certain actions either by themselves or in groups. However, the consequences of the actions in those events can be attributed to agents that were not necessarily the originators of those actions. For instance, if an event represents the actions of an agent A, the consequences of such actions affect that agent only. But if an event represents an agent A talking about an agent B, the consequences of the actions of agent A (the source) also affect agent B (the object), even if agent B is not present in the event.

Therefore, events in a story can be interpreted in different ways depending on the agents that are the focus of the consequences of the actions that occur inside such events. This concept is defined in ISRST as the agent's point of view (POV). Point of views are also useful to study the preferences of the user towards a given agent based on interest and approach interactivity from a more personal and involving perspective.

Even though agents act and develop in the context of events, their points of view are better defined not by the content of the events themselves, but by the relation that joins such events. This perspective gives us the possibility to appropriately generalize the event object and to create suitable interaction points that can be used by the ISRST engine to determine the most interesting way to shape the story. In ISRST we consider relations as directed because this helps us recognize which event constitutes the explained relation of which event. In this context, our representation of a relation  $R$  between two events  $A$  and  $B$  from the point of view of an agent  $X$  is given by the construction:

$$A \xrightarrow{R(X)} B \quad (1)$$

which is read as: The event  $A$  is the  $R$  of the event  $B$  applied to (i.e., being a direct participant of the events) or about (i.e., being an indirect participant, not in presence, but in reference) an agent  $X$ . In this case:  $A$  is considered the origin of the rhetorical relation  $R$  and  $B$  is considered the destination. For instance, if we have two events  $A$  and  $B$ , which are related by the **BACKGROUND** relation with an agent  $X$  as the parameter of the relation, this means that event  $A$  contains background information about event  $B$  involving agent  $X$ . With this approach, not only are we able to have several background relations for a specific event, but it also gives the narrative engine the opportunity to take into consideration the interest of the user when selecting events that are associated to relations with particularly interesting agents.

### ***Nucleus and Satellite Events***

Any story is normally constructed from what Bordwell (1986) referred to as a *fabula* and organized into two types of information: *syuzhet* (main plot events) and "excess" (events that contain extra information). Some of this extra information contained in the story is frequently used to add narrative tension and stimulate the interest of the user in different ways. Therefore, it is plausible to assume that any extra information can be considered a "satellite" (optional) to the important information or main plot, which constitutes the "nucleus" (mandatory).

In the context of RST-based analyses of texts, nucleus and satellite components of a relation are defined to identify which information

is absolutely necessary (nucleus) to understand the text and which information is additional (satellite) and can be omitted without losing the meaning. By performing an extrapolation of these concepts into ISRST, we allow story authors to organize their content based on its importance to the development of the story. For instance, optional content can include the development of events based on different POVs, so that users can experience a story from different agent's perspectives. However, story authors do not need to provide all the possible variations, but only the ones they think might contribute to advance their story in the desired direction. The nucleus and satellite specifications are exclusively applied to events and scenes.

Nucleus events (or scenes) must always be presented in a story, but in the case of satellite events (or scenes), the engine must decide which events (scenes) are to be presented according to the user's level of interest. In general, a good story must include the necessary nucleus information to fulfill its objective as a story and enough satellite information to provide variety for the event selection process.

### **The ISRST Ontology Model**

Taking into consideration the conceptual ideas described in the previous section, we identified the objects that were to be considered in the ISRST model in order to create our semantic storytelling framework. As a result, we created our ISRST ontology model with the following main classes:

- *Concept*: A concept defines a specific topic presented in a story or part of it.
- *Event*: An event is defined as a single piece of meaningful information worthy of being presented. See "The Presentation Module" subsection for more details on how events are constructed in ISRST.
- *Relation*: A relation is the rhetorical binding between two entities (event or scene), which refers to a specific semantic meaning.
- *Agent*: An agent is an entity in a story that is used as an interest target.
- *Origin and Destination*: This pair of classes defines the position of a determined event or scene in a relation.
- *Scene*: A scene is defined as a graph-like structure composed of relations. This class defines the minimum level of organization in which a story arises.
- *Episode*: An episode is defined as the associating object between a scene and a concept.
- *Story*: A story class frames the whole story and is composed of one or more episodes.



the ability of the engine to have more variety in terms of event selection. For a more detailed explanation about the functional characteristics of SRST, refer to Nakasone and Ishizuka (2006).

To overcome these inconveniences in ISRST, we designed a more flexible event organization structure called scene which is based on the capability to associate any two events by using a relation, including in it as many relations as desired. The scene structure was implemented as a directed graph in which each node represents either an event or another scene (inside an origin or destination object) and each link represents the relation that joins any two of these nodes. Figure 7 shows the logical design diagram for the scene structure.

This structure provides not only a convenient way to implement the majority of the conceptual ideas described previously, but also a very flexible organization capable of supporting a highly dynamic story event sequencing process.

A scene is primarily composed by one or more relations, which are the objects that represent the relation concept defined in the “Rhetorical Relations” and “Agents and their Role in Semantic Relations” subsections. To fully identify these relation objects, we defined four attributes that must be specified for each one of them:

- *The Relation’s Name*: This indicates the type of relation that links events or scenes (e.g., BACKGROUND, CAUSE, etc.).
- *The Relation’s Directionality*: This is determined through the specification of the origin and destination components. These objects can reference as their content either an event or another scene, which provides construction flexibility through the use of recursion.
- *Nucleus and Satellite Properties*: Both the origin and destination objects must define if their content is of nucleus type (mandatory) or satellite type (optional).

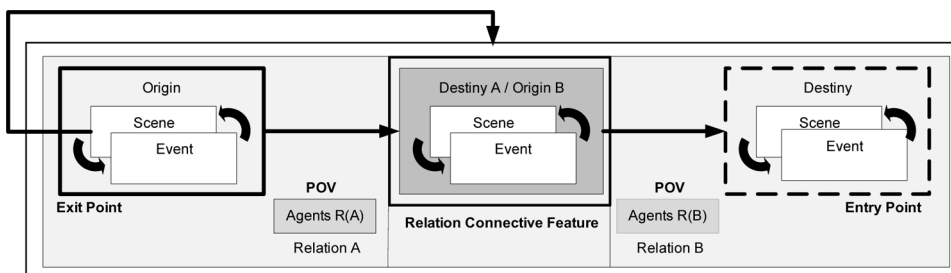


FIGURE 7 ISRST scene structure – logical design diagram.

- *The Relation's Agency*: A relation object must specify one or more agents as POV attributes of the relation (e.g., Agents R(A) and Agents R(B) in the diagram).

In order to connect two relations in a scene, the origin component of one relation must match the destination component of the other. By using this connective feature of the scene structure, we can generate complex links of relations to describe stories. Connections between relations are important for story coherence and event sequencing, which is performed using the narrative concept of advancing the story, explained in the "Qualities of Narrative in ISRST" section.

Additionally to the relation list, a scene makes use of two other attributes that help the storytelling engine deliver the scene content: the entry point and the exit point attributes.

The entry point specifies the first event (or scene) that is going to be delivered at the beginning of the scene. This definition does not imply that the scene must begin with any specific relation, but only defines the starting position from which relations will be selected when the storytelling process commences. The entry point also defines the agents to which the effects of the first event (or scene) will be applied. This allows the story author to define a starting position in terms of agent empathy so that ISRST may be able to begin the computation of the user's interest based on this first assumption.

In contrast, the exit points are defined exclusively by the events (or scenes) in which the scene may end. As a rule, each scene must contain a unique entry point, but may have one or more exit points. This characteristic of the scene structure allows the creation of different storylines inside the scene itself, giving the story author enough flexibility in terms of alternative content selectiveness.

### *Story, Episodes, and Concepts in ISRST*

Even though the flexibility of the scene structure allows us to create rather complex interconnections between events and other scenes through relations, this feature is not enough to deal with the story concept network idea because of the recursive nature of the scene structure itself. If concepts were to be implemented at the scene level, a story author might find himself/herself in the position of specifying concepts even for nested scenes, which proves to be impractical and does not reflect the true meaning of what a story concept is.

Therefore, in addition to the scene object, the ISRST model defines the story, episode, and concept objects that establish the framework for the creation of scene-based stories and give support for the specification of concept networks. In ISRST, a concept object is uniquely associated with

an episode object, which, in turn, contains one scene object. Although having multiple scenes objects per episode is theoretically possible, it is not practical since the recursive nature of the scene object itself allows story authors to have the same effect in their implementations. The advantage of using an episode object as a proxy between the concept and scene objects lies in the fact that an episode not only clearly specifies the boundary between concept and content in a story, but also demarcates its emotive boundaries, which is useful when dealing with content authoring and interest associations to agents.

### **Qualities of Narrative in ISRST**

So far, we have discussed and defined the components that are necessary to create a real semantically conditioned storytelling model. These components only specify how a story is to be organized, but not how a story is to be presented or how narrative principles can be applied in order to convey the content of the story structure in a narrative manner. In this section, we will discuss the two principles implemented in ISRST that add the narrative quality to the model: the concept of advancing the story, which deals with the transition process inside a relation, and the concept of relational narrative template, which deals with the transition process between relations.

#### *The Concept of Advancing the Story*

Most rhetorical relations do not clearly imply a temporal attribute between the events they relate (e.g., the background event is not necessarily shown before the event that refers to it). Therefore, we defined a concept that would allow us to make a sequence, in terms of time, of the events that are joined by rhetorical relations known as advancing the story. The purpose of this concept is to allow the story engine to move through the graph structure created by all the relation objects in a particular scene, establishing the conditions by which the engine can traverse from one component of the relation to the other.

Given a relation specified by

$$A \xrightarrow{R(X)} B,$$

we define the concept of advancing the story through the relation  $R$  if the story's current position, being located at the event/scene  $B$  goes to show the contents of the event/scene  $A$  through the relation  $R(X)$  and moves the story's current position to  $A$ . In other words, at any moment in the presentation of the story, ISRST analyzes all the incoming relations to the current event/scene (i.e., the relations that have the event/scene specified

as their destination component), and moves to the origin of the relation that includes the event that will be shown next. This concept may seem contradictory since ISRST appears to be going backwards in the relation, but we defined this way of processing because:

- It gives a unified model for creating relations in a story. For instance, the CAUSE relation is normally understood as a relation that goes from A to B, but the ELABORATION relation is better understood as a relation that goes from B to A. Since different relations have different ways in which their directionality can be interpreted, the definition of a single advancing paradigm simplifies the understanding of this concept.
- Not all relations follow the pattern of advancing previously described. For example, the CAUSE relation indicates that event A is the cause of event B, but does not necessarily imply that the story moves forward when going to A. In these special cases, ISRST presents the destination event (depending on the user's interest) without moving the story's current position and look for an advancing relation to move forward the story. If we were to specify any relation by its pseudo-temporal attribute, we might end up having a nonuniform model for content creation and, from our point of view, it is much better if these particularities are dealt by ISRST and not by the story author. Authors can benefit from having a unified way to create relations for their story events, since the support from ISRST is totally transparent to them.

Since not all rhetorical relations have the capacity of moving the story forward, we also used the concept of advancing the story to categorize our relations based on this ability. In ISRST, we defined a relation as advancing when it allows the flow of the story to continue from the destination to the origin of that relation, and as nonadvancing when, even though the event or scene in the origin has already been presented, the story's current position remains the same. Out of the nine rhetorical relations considered in ISRST, all but CAUSE and BACKGROUND were defined as advancing.

### *The Concept of Relational Narrative Template*

In a general sense, stories follow a fixed narrative structure that consists of an introduction or explanation of the current state of affairs, a conflict or a change of such state, a resolution of the conflict, and a conclusion, and this is the template agreed upon by the majority of narratologists (Bordwell 1986). In order to integrate this concept of narrative template in ISRST, we also performed a classification of our set of rhetorical relations to determine which ones belong exclusively to specific phases.

In addition, we also took into consideration the fact that each phase must allow the proper relations for the story to advance, and therefore, not



go against the concept of advancing the story. Hence, our implemented narrative structure covers the following narrative phases:

- *Story Introduction*: Only the BACKGROUND relation is used to specify the introduction of a story. The complexity of an introduction phase stems from the fact that a background relation can link either an event or a full scene structure, which may have its own background. The nonadvancing quality of the BACKGROUND relation defines precisely what a story introduction is compared to the idea of moving the story forward.
- *Story Conflict*: Only the CONTRAST relation is considered as the initiator of conflict, or the crisis introducer. Again, the contrast relation can associate either two events or two scenes for more complex constructions.
- *Story Resolution*: Only the SOLUTIONHOOD relation is considered as the initiator for the path to the climax in the story.
- *Story Conclusion*: Only the EVALUATION relation is considered as the initiator for the final conclusion or epilogue of the story.

The remaining relations are classified as generic relations. They can be included in any phase of the story and be used to define more complex structures in terms of relations inside each phase. These relations are SEQUENCE, ELABORATION, PURPOSE, RESULT, and CAUSE.

### **Interest-Based Interaction in ISRST**

Interactivity in storytelling has been implemented in several different ways. Some applications regard interactivity as a simple set of menu choices that are presented during the story, while others make use of more complex hardware in order to give users a true immersive experience. This view of interactivity primarily concentrates on what users can do and manipulate inside the environment of the story, but not on the layout of the story itself.

In contrast, the type of interactivity that ISRST implements takes the view that users should be able to “select” which events to see in the story based on their interest as the story progresses. Interest, being a long-lasting emotion, can shape the sequencing of the story in terms of what the user believes to feel at a particular moment. Even though a story can elicit several kinds of short-lived emotions like joy, anger, and so on, interest remains as the sole emotion that can instill in a user the sense of correctness of a story from his/her own perspective (Tan 1996). In ISRST, interest-based interaction was designed to fulfill two major aspects of interest evaluation in storytelling:

- *Overall Interest Evaluation*, based on Zillmann’s hedonistic theory of the affect regulation function of entertainment, measures general levels

of user interest for the whole story experience (Zillmann 1988). The hedonistic theory states that in the case of extreme understimulation (equated with too little interest in our system), users display certain preference for potentially arousing stimuli and in the case of extreme overstimulation (equated with too much interest in our system), the reverse effect is perceived. In other words, the interest of the user should be kept inside some boundaries in order to not disturb his or her experience in the story.

- *Agent Interest Evaluation*, which is based on the fact that the processes of identification or empathy towards a character involved in the viewing of a story make up a large proportion of the determinants of interest. Since character (agent) empathy is one of the major components of the user's interest in any story (Tan 1996), ISRST isolates this special case from the overall evaluation to properly emphasize the contribution of the user's affective interest. This type of interest evaluation has almost no effect when there are no empathetic characters present. Thus, in the case of storytelling applications whose objective is to provide information in the form of presentations, only the overall evaluation is considered for practical purposes.

In ISRST, the measure of interest is done in a rather direct way by asking users to directly input their interest to the application stored in the form of a numerical value whose range goes from 0 (the lowest value of interest) to 10 (the highest value of interest). Since ISRST was conceived to be deployed as a web application from the beginning, the acquisition of this value is performed using hardware normally available to a web user (e.g., keyboard or mouse). This hardware constraint can be seen as a limitation for its obtrusiveness, compared to other more sophisticated methods that try to detect the interest of the user, like the use of bio-signals (Nakasone, Prendinger, and Ishizuka 2005), or gaze tracking (Bee, Prendinger, Nakasone, André, and Ishizuka 2006). Despite this fact, we concluded that our method to collect interest values is the most appropriate for the model because of the following:

- The use of bio-signals to determine user interest is rather ambiguous since real-time assessment of physiological data has several problems, such as finding an appropriate baseline (i.e., a condition against which physiological change can be compared), and how the intensity of an emotion, in this case interest, is reflected in the physiological data (Levenson 1988).
- Even though visual attention, accurately detected by gaze-tracking algorithms, is highly correlated to some aspects that define interest in a user, the interest defined for stories is more closely related to

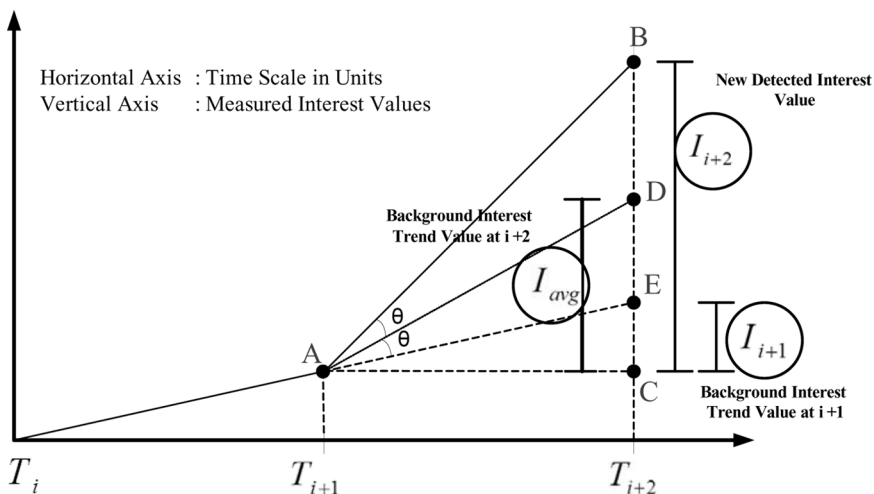
the emotional aspect than to the visual attention, especially when this interest is directed to a specific entity in the story (e.g., an agent).

- Unlike emotions like joy, sadness, and the like, interest is a rather long-lived emotion that can be accurately measured by direct questioning to the user (Tan 1996). Storytelling systems acknowledge this fact when they provide obtrusive interfaces like menu selection or voice commands, assuming these interfaces have diegetic relevance.

In subsequent subsections, we will describe in detail the two types of interest evaluation performed by ISRST.

### ***Overall Interest Evaluation***

According to the first principle of interest proposed by Tan (1996), momentary values of interest must be influenced by the stimulus elements presented earlier in the story, but only to the extent that these are available in the memory of the user. In other words, the true interest of the user at a particular point in a story depends not only on the current interest value at that point, but also on the contribution of previous interest values to the experience. Previous interest values, commonly known as background interest, define their contribution, not as an isolated set of values, but as a trend that indicates the evolution of the user's interest throughout the whole story experience. When a new interest value is detected from the user, a new background interest is calculated based on how this value affects the previous interest trend. A schema on how this calculation is performed is shown in Figure 8.



**FIGURE 8** Background interest trend calculation.

At some time  $T_{i+1}$ , a background interest trend value, defined as  $I_{i+1}$ , represents the averaged trend value of the contribution of previous interest values that range from  $T_0$  up to  $T_{i+1}$ . When a new interest value,  $I_{i+2}$ , is detected from the user at time  $T_{i+2}$ , the new background interest trend value at that point ( $I_{avg}$ ) is obtained by calculating the slope average between  $I_{i+2}$  and  $I_{i+1}$ . In this case, the slope average accurately reflects the real contribution of the new interest value by modifying the background interest, not in magnitude, but in orientation. According to Figure 8, the new background interest trend value is calculated using the following formula (having constant time intervals):

$$I_{avg} = \tan\left(\frac{\arctan(I_{i+1}) + \arctan(I_{i+2})}{2}\right). \quad (2)$$

Geometrically speaking, this value represents the distance CD formed when the line that bisects the angle formed by both the old background interest value and the new interest value (angle BAE) intersects the segment CE. In ISRST, we call this value the overall interest value, and it is calculated every time a new interest value from the user is input in the application.

#### ***Agent Interest Evaluation***

Interest values are not only influenced by previous stimulus presented in the story, but also by expectations about what is to come to the extent that the expectations are based on elements already received and are actually active in the user's memory. These active elements are represented by the characters or agents that take part as actors in each event, since empathetic emotions, such as interest, are characterized by the valence of the events in relation to the concerns of the characters (Tan 1996). Therefore, it is plausible to assume that if the user has a certain level of interest, while a particular set of agents bear the consequences of the actions in the event that is currently being shown, that level of interest can be attached not only to the overall feeling of the story, but also to each of the affected agents.

In order to implement this feature in ISRST, each agent is associated to a value that indicates the overall absolute average of interest calculated by averaging all the individual interest values that the user conveyed when that agent was active (i.e., was being shown or was bearer of an action's consequences) in the story. Unlike the overall interest evaluation case, the agent interest evaluation is performed on absolute numerical interest values and not on trends, since what determines the user's interest in an agent is not dependant on the smoothness of the flow in the story, but on empathetic emotions (e.g., sympathy for the agent) which tend to be more fluctuating.

In ISRST, we call this value the agent interest value, which is also calculated every time a new interest value from the user is input in the application.

### Telling an Interesting Story in ISRST

The main processes of the ISRST engine were constructed under a principle of strict modularity, which allowed us to have a very clear separation between the visual component of the application and the story ontology reasoning component. This distinction is important in order to have a properly defined functional division and to guarantee the independence of the model itself. Therefore, to isolate the event sequencing process of a story from the actual presentation process, the ontology component of the ISRST engine encapsulates in it all the complexity of the story structure and its status variables and provides an interface whose sole function is to deliver the appropriate next event taking into consideration the changes in the environment of the user and the story.

The diagram of the algorithm for that interface is shown in Figure 9.

In order to select the most appropriate next event, the ISRST ontology component (OC) performs the following procedures and decisions:

- Check if an exit point of the currently active scene has been reached. As part of the set of internal variables that the OC stores is a queue of

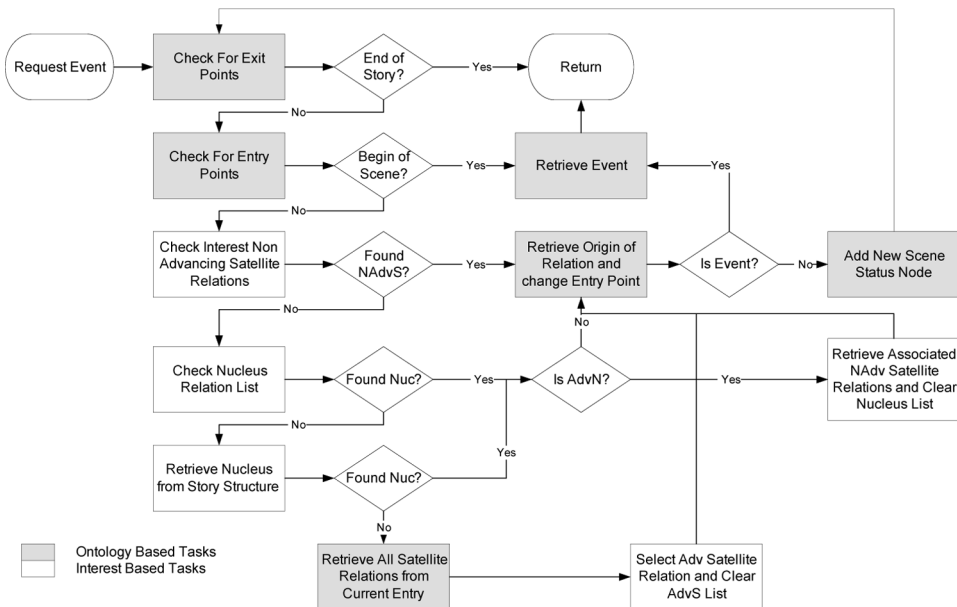


FIGURE 9 The event retrieval function diagram.

objects called `SceneStatusList`, which stores the information about the scenes that the story will show. When an exit point has been reached, the OC removes the correspondent scene object in the queue and searches for its parent scene recursively upwards until it finds one with a current position that is not an exit point itself. If the OC cannot find any parent scene, it means that it has reached the end of the whole story and, therefore, returns a token confirming such condition.

- Check if the current position in the active scene corresponds to the entry point of that scene. If the OC detects that no event from that scene has been previously shown, it retrieves the object indicated by the entry point. If the object is an event, it establishes it as the function's return value. If the object is a scene, the OC stores the information of the currently active scene object in the `SceneStatusList` queue and sets the new scene object as the currently active one. The OC performs this process recursively until a valid entry point in the form of an event can be found. If no event is found, the OC returns a token indicating a story structure error (i.e., invalid entry point in the scene).
- If a valid current position is found (i.e., an event in the scene that has already been shown), the next event is determined based not only on the story structure criteria, but also on the registered interest values from the user. In this context, the classification of relations in *advancing* and *nonadvancing* plays a very important role to determine which content can move the story forward and which content provides additional information to satisfy the hedonistic attribute of the user's interest. By combining the analysis of this classification with the nucleus-satellite property of the components of the relations and the user's interest values, the OC makes a structured search for the next event, giving priority to the interest of the user over the narrative-wise predefined sequencing of the story itself. The steps that the OC performs are:
  - *Check for Nonadvancing Satellite Relations*: The first step in order to select the next event in a story is to analyze the current level of the user's interest, by performing an overall interest evaluation (see the "Overall Interest Evaluation" subsection for details on how to calculate the overall interest). When the overall interest trend average falls inside the range of normal threshold values defined as parameters in ISRST, this step is skipped. But when this average falls outside the range, the OC assumes that the user needs additional information to keep him or her from being overstimulated or understimulated. A relation that contains additional information (nonadvancing satellite relation) is a relation that (1) has a satellite origin entity, and (2) does not move forward the story. These two requirements are necessary for the OC to be able to maintain the coherence in the main story. In order to select the best relation, the list in which this additional information is

stored, called the `SatelliteAdvList`, is ordered by the following criteria in this order: agent interest, relation proximity, and relation template (see the “Ordering Criteria for Relation Lists” subsection for more details on the ordering criteria). Once the list is ordered, the OC picks the topmost relation as its first candidate. The only difference between the understimulation and overstimulation cases lies in the way the list is ordered by the agent interest criterion. For understimulation, it is regarded as more pleasurable for the user if the selected relation contains an agent in which he or she has the most interest in, whereas for overstimulation, it is assumed that a relation with an agent in which he or she has the least interest in could help the user calm down and enjoy other agents of the story as well.

- *Check for Nucleus Relations:* If the overall interest evaluation regarded the user’s interest as stable (i.e., within normal interest threshold values), the OC checks from the current position for any relations that have nucleus origin entities in them (i.e., nucleus relation). The main requirement for the selection of nucleus relations is that they all must be presented, regardless of the user’s level of interest. Therefore, in order to decide which nucleus relations come first, found relations are stored in a list called `NucleusList` and sorted by the following criteria in this order: nonadvancing–advancing, relation template, and agent interest. Unlike the previous case, the emphasis of the order is in the priority of nonadvancing relations over advancing relations and the narrative quality of the relations, since all the nucleus events in the relations must be shown. However, if a nucleus relation that is of the advancing type is found and presented, all other relations in the `NucleusList` list are discarded in order to maintain the coherence of the story structure, which has a higher priority. At this point, before the story moves forward, all the satellite nonadvancing relations that are also found from this current position are stored in the `SatelliteAdvList` list for further reference.
- *Check for Advancing Satellite Relations:* If no nucleus relations were found, the OC looks for advancing satellite relations in order to move the story forward. An advancing satellite relation is a relation that (1) has a satellite origin entity, and (2) moves the story forward. The process for selecting from these types of relations (which are stored in a list called `SatelliteAdvList`) is similar to the process for selecting nonadvancing satellite relations (minus the ordering by the relation proximity criterion). The main difference lies in the fact that, since the selection involves only advancing relations, once a relation is chosen, the other relations in the list are discarded in order to preserve the coherence of the story structure, as in the case of the nucleus relations processing.

- Once a relation has been selected by any of these three procedures, the OC retrieves the origin entity of the relation, and modifies the current position of the scene if the relation is of the advancing type. If the retrieved entity is an event, it is established as the function's return value. If it is a scene, it is inserted into the SceneStatusList queue and the function is called again. If no relation was retrieved, this indicates a story structure error and the function returns a token indicating such condition.

In the next section, we will detail the criteria by which the ordering of the relation lists is performed.

### ***Ordering Criteria for Relation Lists***

Depending on which type of relation list the OC extracts the information from, the list needs to be ordered using a determined criterion, so that the OC can find and select the best relation from that list. The implementation of these criteria in ISRST is based on narrative considerations and the role that interest plays during the storytelling process. The three main ways to order a relation list in ISRST are:

- *Relation Template Ordering*: The relation list is ordered by the sequence of phases defined in the relational narrative template. In other words, the relations that belong to the background phase come first, then the relations that belong to the conflict phase, followed by the relations that belong to the resolution phase, and, finally, the relations that belong to the conclusion phase. The purpose of the relation template ordering is to enforce a narrative quality into the relations in the list. When more than one relation is found for a specific phase, the nonadvancing relations take precedence over the advancing ones.
- *Agent Interest Ordering*: The relation list is ordered based on the registered level of interest for each of the agents that are active (i.e., are being presented) and defined as part of the relation. The OC orders the internal list of agents based first on their status (active or inactive) and second, on their interest average values. This ordered agent list is then used as a template to order the relation list (i.e., the relations which have agents that are ranked first in the ordered agent list come first).
- *Relation Proximity Ordering*: The relation proximity concept is defined as the distance in relation steps that exists between the previously selected relation associated with the current position and any other relation in the list. The definition of relation step is based on the concept of advancing the story. For instance, if a relation in the list was stored before the story advanced N relations (through advancing relations only), then the relation proximity factor for the stored relation is N. This



factor measures the age of the information in the relation and is useful when the OC needs to decide what type of old information the user would be more inclined to accept as part of the natural flow of the story.

We have described the major components and concepts of ISRST and its value in the context of semantic narrative and storytelling paradigms. In the next section, we will present the web application that demonstrates the full capabilities of the ISRST model.

## **THE ISRST APPLICATION: IMPLEMENTING FOR THE WEB**

The ISRST application is a fully functional Java-based system that implements the concepts from the ISRST model. In this section, we will detail the technical features of our ISRST implementation, emphasizing the following aspects: (1) the presentation module, which deals with event specification and presentation on the web, (2) the ontology reasoning module, and (3) additional features of the implementation like a central interest logging service for test studies. You can test the implemented system at ISRST (2009).

### **The Presentation Module**

One of the main objectives in the development of ISRST was to provide a flexible way to present different types of story content using a simple interface that could be deployed on the web for widespread use. For this reason, each major component of the story presentation module was carefully designed or selected to achieve this purpose.

The presentation module consists of a visual applet that displays the story events and additional support Java archives (JARs) that enhance its capability to process different types of multimedia content. Figure 10 shows a screenshot of the application and its main visual components.

The presentation module has four main components:

- *The Main Presentation Area:* In this area, the content of the story is presented to the user. The current version of our application provides support for the following types of content: text, still images, audio files, and video files. In order to provide the story author with a flexible way to make use of these types of media in any possible combination, we developed the ESL, which allows the author to create instruction script files that can generate very complex multimedia events. The functionality provided by ESL includes: (a) audio file playback, (b) image and video display, (c) text display, including support for different languages, (d) simple visual effects for image and text presentation like fading, and (e) media synchronization support through



FIGURE 10 ISRSST application – main screen.

the use of delays. An example of an ESL file is shown in Figure 11. Most of this functionality is supported by the standard Java Virtual Machine (our application was implemented using JDK 6.0) except the audio file processing capability (provided by a customized version of

```

PLAY@@@%HOMEADD%/bgmusic/LauraSullivan_HopeForTheSun.mp3@@@Y
DELAY@@@2000
PICT@@@%HOMEADD%/trailerim/Trailer1.jpg@@@FADE ← Fadein Image Support
DELAY@@@1000
TEXTFADE@@@0@@@30@@@30@@@This is the story of a man who was in search of his
destiny###%SP%Esta es la historia de un hombre en busca de su destino@@@cyan@@@22
DELAY@@@3000
TEXTFADE@@@0
DELAY@@@2000
TEXTFADE@@@0@@@50@@@50@@@Even if that destiny meant to leave his
hometown...###%SP%Aunque ese destino signifique dejar su hogar...@@@cyan@@@22
DELAY@@@3000
TEXTFADE@@@0 ← Instruction Delay Support
DELAY@@@2000
TEXTFADE@@@0@@@300@@@300@@@...And everything else behind...###%SP%...Y todo lo
demas atras...@@@cyan@@@22
DELAY@@@3000
TEXTFADE@@@0 ← Fadeout Support
DELAY@@@2000
PICT@@@CURRENT@@@NULL

```

FIGURE 11 An ESL file sample.

JLayer (2009)) and the video file processing capability (provided by the Java Media Framework (JMF 2009)).

- *The Story Control Buttons*: These buttons are used to start, pause, or stop the execution of the story presentation and provide all the functionality that is needed to control the application.
- *The Dialog Speed Selector*: In some cases, the speed at which text lines are presented can be too fast or too slow to read for certain users, which can alter his or her level of satisfaction in the story and change it into frustration over the technical aspect of the application. To overcome this problem, this control was introduced to let the user select the speed of the text in the story, which ranges from slowest (which is about 60% slower than the normal speed defined by the story author) to fastest (which is about 60% faster than the normal speed).
- *The Interest Indicator Bar*: In this area of the screen, the interest values specified by the user are displayed as a gradient of colors, going from the blue color (which indicates the lowest value of interest) to the red color (which indicates the highest value of interest), but internally, this is translated into our predefined numerical scale (0–10). Interest values are input into the application by moving the mouse wheel up or down; up indicating increase of interest, and down indicating decrease of it. The values are then sampled at regular intervals and sent to the ontology reasoning module to be processed and stored. This method to detect user interest was selected because the display of actual numerical values for interest on the screen may make the user too worried about trying to estimate accurately the level of his or her interest, which could affect his or her experience in the story. A gradient of colors, on the other hand, provides the same functionality exactly, giving a more abstract, but less distracting, way for the user to express his or her feelings.

### **The Ontology Reasoning Module**

The ontology module is in charge of the management of the story ontology and provides the reasoning capability to determine the course of the story based on the interest of the user and the story structure itself. ISRST uses the Kaon2 Reasoning Engine (Kaon2 2009), downloaded as a component of the client part of the application, to deal with the specific manipulation of the ontology files that constitute the storytelling model.

This module is the core component of ISRST and performs the following tasks:

- The initialization process, which includes the ontology file and story instantiation and the application parameter loading.
- Concept path calculation from story episodes.
- Asynchronous recording of interest log entries.

- Event retrieval based on the process described in the “Telling an Interesting Story in ISRST” section.

### **Additional Implemented Features**

One of the objectives of the ISRST web implementation was to conduct a test of the functionality of the application, and compare it against the level of satisfaction of its users. In order to make the implementation of the tests more flexible, we have also developed additional nonfunctional features that helped us accomplish this goal:

- *ISRST Application Parameterization*: ISRST was designed to support a large range of configurable parameters that control the reasoning process in the story structure and the interest measurement. Among the different parameters that can be manipulated are the categorization of advancing relations, the relation proximity distance threshold, the maximum and minimum thresholds for interest values, and the way to choose satellite relations based on interest values.
- *Interest Server Logging*: Even though the application itself runs on the client side, the interest values that are captured from the user are automatically sent to our server, where it is stored in text files. This allows us to assess the performance and usability of ISRST regarding the user’s interest in the application. In this case, we have tried to be consistent with every good practice regarding the ethical use of stored information by informing users that their interest data will be logged and that any personal information will only be used for statistical purposes and not distributed to any external institutions.
- *Questionnaire Logging*: When the user has finished testing our application, a small questionnaire is presented concerning both the content of the story and the nonfunctional aspects of the application itself. Given the long-lasting characteristic of the interest emotion, the use of questionnaires to measure interest has proven to be stable across several conditions of use and provides a good differentiation (Tan 1996).
- *Language Support*: Through the use of ESL, users are able to provide content in different languages using the same story and event structure.
- *OWL Story File Generation Tool*: This ISRST component was developed to help story authors deal with OWL in a much easier way, generating the OWL story file required by the ISRST engine from a simple XML specification file.

### **TESTING ISRST WITH A REAL STORY**

In this section, we will describe the story that was used to test our application in a real environment, along with the results we have obtained.

### **“When Your Heart Takes Over” – A Real Story Created for ISRST**

“When Your Heart Takes Over” (WYHTO) is a dramatic story created for the ISRST application in order to (1) test the full potential of the ontology reasoning module in a real setting, and (2) provide our testing community of users with an engaging and well-constructed story, based on a solid premise and richly developed characters (Egri 1972). The main objective for the creation of WYHTO is to test the ability of the engine to utilize the registered interest values from the user to select an outcome that is the most satisfying and pleasant for him or her in terms of affective closeness or sympathy for the character included in that particular outcome.

“When Your Heart Takes Over” is a story that offers episodic emotion sequences combined with dramatic event intensity which will encourage the user to progressively develop feelings towards the protagonists and, consequently, channel his or her interest in the content of the story in a more precise way. The creation of complex stories like WYHTO demands certain requirements and steps to be performed in order to ensure not only its compatibility with ISRST, but also its coherence as a story from a thematic point of view (Tan 1996). These are the issues that were considered during the creation of WYHTO:

- The thematic approach was devised from the point of view of a traditional feature film, which implies a happy ending and the enforcement of intentional behavior in the protagonists (Bruner 1991).
- A clearly defined episodic organization, which was used to regulate emotions around particular agents in the story. WYHTO consists of four episodes that provide a full 90-minute story experience. This story structure gives us the possibility to unambiguously associate episodic interest with a particular agent in order to determine the satisfying outcome of the story in a precise way.
- A clear identification of the preference of the user in terms of story content. In our case, the outcome of the story was designed taking into consideration the preferences associated with character empathy that might dwell in the user.
- The generation of empathetic emotions towards the characters. In WYHTO, this empathetic emotion in the form of sympathy not only stimulates the user selection for a preferred outcome, but also makes it possible for ISRST to objectively ask for his/her level of interest at any moment in the story.

Figure 12 shows part of the logical design for the first episode of WYHTO, highlighting the structural aspects of it. In this diagram, events

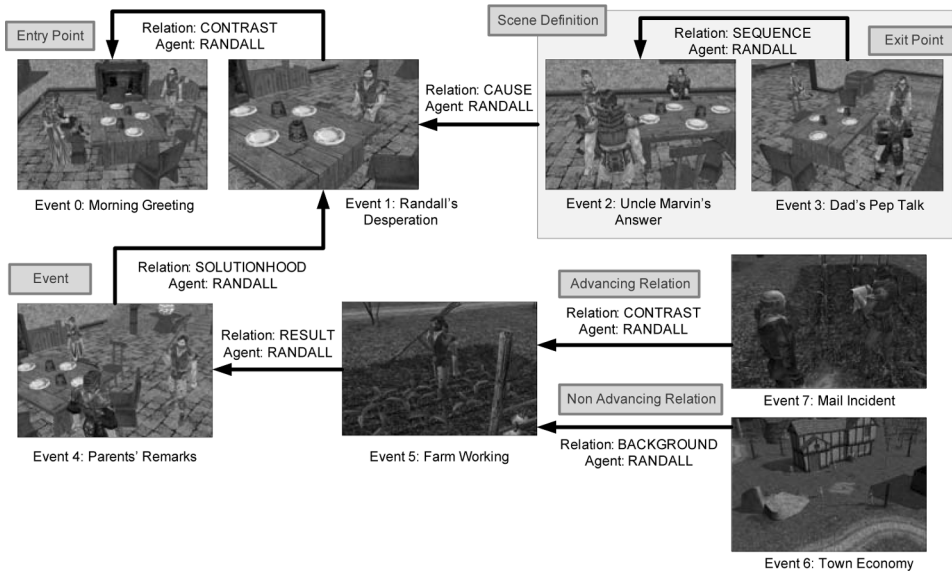


FIGURE 12 WYHTO Episode 1 – logical design diagram.

are represented by pictures from the actual story and relations by directed arrows. Entry point events (i.e., events 0 and 2) indicate the events from which the scene will start. Exit point events (i.e., event 3) indicate the events in which the scene will end. The relation label over each arrow describes the type of relation and the agent(s) that are affected in it.

According to Lang (1984), dramatic presentations are more effective in producing affect than spoken text. Therefore, to provide a suitable artistic environment to channel the feelings of the users more convincingly, WYHTO events were implemented using the full multimedia capabilities of the presentation module:

- Events were composed of images that are laid out using some visual effect techniques like fading.
- Character dialog synchronization allows a more dramatic effect in terms of timing.
- Background music helps create in the users a better mood to perceive the emotional intention of the characters in the story.

### Story Data Analysis

“When Your Heart Takes Over” was purposely designed to enable us to perform a test study in which we could analyze the correlation between the interest of a user in an agent (i.e., agent empathy) and the level of satisfaction in the story given an outcome which emphasizes that particular

agent. Even though we have to be careful with drawing conclusions from a study with a small number of participants (10 subjects), the application demonstrated its capacity to deliver interesting stories. This could be shown by comparing the experience of a group of people that actually tested ISRST in its entirety and gave us their extremely valuable comments. Creating interest in the user is not only a matter of crafting a good story, but also giving the story the proper structure so that interest can be aroused, accurately associated to agents, and discerned in order to use it to enhance the experience in the story itself. Therefore, the four-episode structure of WYHTO was defined in order to deal with each of these three aspects:

- Providing an episode for interest stabilization (Episode 1). In this episode, the user is encouraged not only to get to know the background of the story, but also to familiarize with the method by which his or her interest is to be measured. This episode is the longest and provides a way for the user to relax and feel comfortable. This episode is composed of 18 nucleus events and 17 relations, including 1 subscene.
- Creating two episodes for interest buildup (Episodes 2 and 3). In the next episodes, the interest values of the user start to be practically associated with agents. In WYHTO, the interest is associated with mainly two agents, so each episode deals with one particular agent at a time. Since each episode deals only with events that are related to one agent, this allows us to precisely associate the interest values generated in each episode to that agent. Episode 2 is composed of 11 nucleus events and 10 relations, including 1 subscene, whereas Episode 3 is composed of 14 nucleus events and 13 relations, including 3 subscenes.
- Providing an episode that gives a satisfactory outcome based on the interest values measured in the two previous episodes (Episode 4). This final episode provides the climax of the story whose events will be determined based on the preference that the user showed toward one agent or the other and be fulfilling from the point of view of story satisfaction. This episode is composed of 2 nucleus events, 10 satellite events, which correspond to the different outcomes in the episode (5 satellite events per outcome), and 11 relations.

For this study, we created two scenarios in order to identify when a high correlation between empathy and satisfactory outcome is more likely to be present:

- *Scenario A*: The outcome of the story reflects the choice of the user; that is, the agent presented in the outcome episode of the story is the one in which the user showed more interest during the previous episodes.

- *Scenario B*: The outcome of the story goes against the choice of the user; that is, the agent presented in the outcome episode of the story is not the one in which the user showed more interest during the previous episodes.

The hypothesis in our study was that the satisfaction of a user could be measured by computing his/her interest values during the presentation of the outcome episode of the story. In other words, the averaged interest value calculated for the user during the outcome episode would be higher if the outcome episode were fulfilling his/her expectations in terms of empathy than if not. The previous three episodes of the story are only to provide the users with the background for character empathy build-up and interest direction towards one of the two possible agents.

Our study followed a between-subjects design and was carried out with a group of 10 people divided into two subgroups of five people each. All of them were university students from computer science-related courses whose age ranged from 19 to 30 with almost no experience in digital storytelling. Each of the subgroups contained two female students and three male students. One subgroup tested the application using the Scenario A parameter setting and the other using the Scenario B parameter setting. Neither of these groups was aware of this discrimination in order to have more objectivity in our study. The statistical information regarding detected user interest values during the outcome episode is shown in Table 1.

By performing a one-tailed, *t*-test analysis on these sets of interest values (in the range [0, 10]), we found that the statistical difference between the interest values of the two groups was significant ( $p < 0.01$ , *t* Stat = 2.9524). Despite the small size of the sampled data, this finding supports our premise that people might generally have more interest in a story if its outcome presents a favorable situation with a character with which they built an empathetic relation.

### ***Questionnaire Results***

In order to confirm the results obtained by the interest value analyses, we compared the values obtained from our questionnaires in both

**TABLE 1** Statistical Information Regarding Detected User Interest Values During the Outcome Episode

Scenario	Mean	$\sigma^2$	MIN	MAX
Scenario A	6.6055	0.6901	5.6152	7.6779
Scenario B	4.9204	0.9387	4.1465	6.2561



scenarios. The main set of questions formulated to the user was:

- *Overall Enjoyment of the Story*: In this question, the user was asked to input his or her level of enjoyment of the story in terms of theme and technical construction. The range of values went from 1 (boring) to 5 (exciting).
- *Appropriateness of the Length of the Story*: In this question, the user was asked to assess the length of the story. The range of values went from 1 (too long) to 5 (too short).
- *Outcome Satisfaction*: In this question, the user was asked to input his or her level of satisfaction with the selected outcome episode. The range of values went from 1 (satisfied) to 3 (not satisfied).
- *Presentation Style*: In this question, the user was asked to give his or her appreciation regarding the usability of the application's interface. The range of values went from 1 (too simple) to 5 (too complex).
- *Interest Bar Usability*: In this question, the user was asked to rate the usefulness of the interest bar. The range of values went from 1 (perfect for interest input) to 5 (interest measurement was not important).

Table 2 shows the statistical information regarding the questionnaire answers in both scenarios.

The application of a *t*-test analysis did not give statistically significant results, but it provided us a good reference on the qualitative assessments of the participants. It can be noticed that the story enjoyment value in Scenario A differs from almost one point to the value in Scenario B, which could indicate that a preferred outcome may help users enjoy a story more in the overall sense. Nevertheless, the fact that both groups still claimed to enjoy the story suggests that enjoyment may not only depend on agent empathy, but also on issues like the topic of the story and the content implementation techniques (e.g., event pictures, background music, dialog performance, etc.), which were not considered in detail for this study. On the other hand, the outcome satisfaction values present a less distinctive difference. Although these results suggest that character empathy detected through interest does have certain influence in the

**TABLE 2** Statistical Information of Answers for Scenario Comparison

Questions	Scenario A				Scenario B			
	Mean	$\sigma^2$	MIN	MAX	Mean	$\sigma^2$	MIN	MAX
Story enjoyment	4.8	0.2	4	5	3.8	0.7	3	5
Story length	2.6	0.3	2	3	2.8	0.2	2	3
Outcome satisfaction	1.2	0.2	1	2	1.6	0.3	1	2
Presentation style	2.8	0.2	2	3	2.2	0.7	1	3
Interest bar	1.8	1.7	1	4	2.2	2.7	1	5

satisfaction of the user, a study involving more subjects is needed to verify this claim. In addition, we could also observe the following: (a) users considered the story length to be either okay or somewhat long, which is encouraging for storytelling applications that seek to provide long experiences over the web, (b) users appreciated the simplicity of our application interface, and (c) users felt that our interest bar could be complemented with other types of input devices, but accepted this control as the primary method to input interest in these kinds of systems.

In the comments section of the questionnaire, some participants pointed out that “the quality of the graphical representation for the characters was a bit disappointing,” indicating that the satisfaction of these users might have been reduced due to the graphical aspects of the story implementation. Improvements on these technical issues had to be sacrificed in order to have an adequate performance for ISRST to be deployed on the web.

## **DISCUSSION AND LESSONS LEARNED**

During the implementation and testing of ISRST, we faced technical and usability issues which, in some cases, contributed to enrich our application, and in others, limited its capability to deliver a more fulfilling storytelling experience:

- When we were designing the visual component of ISRST, we decided to implement a simple interface where the area used to display the story content would occupy the maximum space possible and the controls to manipulate the story playback would be reduced to the minimum set. This was done to let users concentrate on the content of the story itself, without adding the frustration to manipulate complicated control parameters. According to the questionnaire answers, this design was very much appreciated.
- Taking into account the suggestions of some early users of ISRST, we added two extra controls to an earlier version of our interface: the dialog speed selector and the pause control button. The dialog speed selector was included to regulate the dialog text display time, so that users would be able to adjust the duration of each piece of text on the screen (not to control the speed of the story content itself). Missing dialog text in the story led some testers to get frustrated and uninterested. The pause control button was included to let users have a respite during the testing, since some people expressed concerns about having to interact with the application for almost 1:30 hours. This button was included to let users have brief pauses, but not long ones. Although interest is a long-lasting emotion, having too long a pause could actually affect their perception of the story in terms of interest.

- Since we only used static images to display the content of our story, some test participants conveyed their minor disappointment at the application, citing that the story was not so attractive due to the lack of emotion expressiveness in the characters. During the implementation of WYHTO, we considered adding video shots and text to speech (TTS) processing to try and overcome the emotional issue, but although these features were implemented in ISRST, we could not deliver them as part of our story. In the case of the video shots, we prepared a test file of 5–10 seconds and tried to display it in ISRST. Due to codec constraints in the Java Media Framework software, the best encoding we could use gave us a 5–8 Mb file, taking almost 10–20 seconds to load in a fast network connection, which was prohibitive as a lag. In the case of the TTS processing, we tested our system using a Loquendo TTS Demo web page (Loquendo 2009) as an online processing source for speech generation. Unfortunately, the resulting speech lacked the type of emotional tone that we needed, so we preferred to leave this option out as well.
- Since the main objective of ISRST was the implementation of a storytelling engine, the creation of a story content authoring tool was not contemplated in the beginning. Therefore, most of the content had to be created manually. The implementation of WYHTO, which included the argument, the dialog script, the storyboard, the image shots, and the post-production (shot sequencing, text synchronization, and background music selection) took approximately 3 months. The playback of our original story lasted 2:30 hours, so it had to be cut in order to have an acceptable time for testing, but without reducing too much of the emotional quality of the situations created to induce empathy in the user.
- ISRST has not been given to external people to develop their own stories yet, so we do not have an assessment regarding authorship flexibility for now. However, we can argue, based on our own experience making WYHTO, that the easiest way to create a story for ISRST is to follow these steps: (a) define which shots or combination of shots would constitute the events, (b) specify the semantic relations between those events, (c) identify the events that would be the most likely candidates to conform a subscene, and (d) select which events have content that could be considered as optional. This requires some preplanning in terms of conceiving stories as discrete related events, which can be considered the most critical part during story design. In addition, the keys for story variability using the exit point feature lie in choosing the story branching points carefully (i.e., points during which ISRST will decide which direction to take based on detected user interest values) and correctly labeling their associated events as satellite (optional). Using this approach, the construction of the different alternatives in our story was performed without problems.

- General comments from the test participants indicate that the addition of background music was definitely a plus and helped to add a more involving atmosphere to the story.
- The main reason behind the creation of ESL was to provide a flexible way to define and synchronize story content outside the proper definition of the story ontology, since no content authoring tool was implemented. By separating story content from story structure creation, we gave authors the possibility to use standard OWL tools such as Protégé (2009) to create their story structure, and ESL to create the content itself.
- Although OWL tools currently provide user-friendly interfaces to manipulate OWL content, we realized that, even with these tools, story authors with little experience in semantic computing could face some inconveniences in trying to accurately specify their stories using such tools. Thus, we developed a component in ISRST to help authors generate their OWL story files from very simple XML files containing the structure of their stories.

## CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

In this article, we have presented ISRST, our proposal for a real semantically conditioned storytelling ontology model based on the organization of events using a meaningful subset of relations proposed by the RST and narrative principles applied to these RST relations combined with interest measurement to provide a unique interactive storytelling experience.

Appropriately selected features give the application a solid base as a storytelling framework and flexibility in story implementation including the following:

- The utilization of OWL class structures to define major story components. Although the current functionality is limited to object searching and attribute extraction, the use of this semantic tool gives ISRST the possibility to further encapsulate the model by implementing more complex reasoning inferences involving already implemented classes and real-time parameters. In addition, OWL provides to the story author a well-known standard way to create and implement his/her story structures.
- The construction of complex multimedia content using ESL. Event specification language provides not only the way to personalize this content, but also a useful way to synchronize it to enhance a story's presentational effects. Story authors can, thus, specify their whole stories using OWL for story structure and ESL for story content.

- The ease of web deployment by implementing the whole application in Java.
- The implementation of a true unobtrusive interaction model through the use of interest, which not only is the most important factor in any user interaction paradigm, but can also be extrapolated to other well-known types of interaction methods such as menu selection, gamepad manipulation, command processing, action perception, etc.

In order to improve ISRST functionality and usability, the following issues have been identified:

- *Comprehensive Test Studies:* We would like to corroborate the results obtained in our pilot study by conducting an extended study using more participants. In this case, we would also like to include in our questionnaire inquiries about interface usability, application performance, and story content quality, which will help us know the technical concerns of the users in a more formal way.
- *Optimal Parameter Configuration:* Fulfilling story experiences can be associated to several conditions other than the analysis of character empathy from the user's perspective. In this context, we would also like to perform empirical analyses by modifying the following:
  - The threshold values for overall interest trend averages. The current version of ISRST has fixed values for these thresholds, which were established based on empirical studies on interest in feature film presentations (Tan 1996).
  - The ordering of agents based on interest. In a normal scenario case, ISRST was designed to deliver story content based only on the maximum value of agent interest, as in WYHTO.
- *Content Authoring Tool:* We would also like to create a tool that would allow story authors to define in a more graphical way the organization of their story structures and the synchronization of their multimedia content.

In general, our model design and implementation were conceived to be as highly configurable as possible, even though it had restrictive assumptions that needed to be established for technical and story content-related reasons (e.g., fixed threshold values for interest calculation, predefined narrative order, and a reduced set of story ontology classes and relations). Despite these limitations, we are confident that our contribution will be most significant in the quest to add narrative quality to general applications which seek to stimulate in their users better understanding and assimilation of their presented information.

## REFERENCES

- André, E., K. Concepcion, I. Mani, and L. van Gulder. 2005. Autobriefer: A system for authoring narrated briefings. In: *Multimodal Intelligent Information Presentation*, eds. O. Stock and M. Zancanaro, pp. 143–158, New York: Springer.
- Bee, N., H. Prendinger, A. Nakasone, E. André, and M. Ishizuka. 2006. AutoSelect: What you want is what you get. Real-time processing of visual attention and affect. In: *Proceedings International Tutorial and Research Workshop on Perception and Interactive Technologies (PIT-06)*, Springer LNAI 4021:40–52.
- Bordwell, D. 1986. *Narration in the Fiction Film*. London: Routledge.
- Bruner, J. 1991. *Acts of Meaning*. Cambridge: Harvard University Press.
- Callaway, C. and J. C. Lester. 2002. Narrative prose generation. *Artificial Intelligence* 139(2):231–252.
- Callaway, C., E. Not, A. Novello, C. Rocchi, O. Stock, and M. Zancanaro. 2002. Automatic cinematography and multilingual NLG for generating video documentaries. *Artificial Intelligence* 165:57–89.
- Cavazza, M., F. Charles, and S. J. Mead. 2002. Interacting with virtual characters in interactive storytelling. In: *Proceedings of the First Conference on Autonomous Agents and Multiagent Systems (AAMAS-02)*, pp. 318–325. Bologna, Italy: ACM Press.
- Crawford, C. 2005. *Chris Crawford on Interactive Storytelling*. New Riders.
- Csikszentmihalyi, M. and I. S. Csikszentmihalyi. 1988. *Optimal Experience. Psychological Studies of Flow in Consciousness*. Cambridge, UK: Cambridge University Press.
- Egri, L. 1972. *The Art of Dramatic Writing*. New York: Touchstone.
- Elliott, C., J. Brzezinski, S. Sheth, and R. Salvatoriello. 1998. Story-morphing in the affective reasoning paradigm: Generating stories semi-automatically for use with “emotionally intelligent” multimedia agents. In: *Proceedings of the Second International Conference on Autonomous Agents*, pp. 181–188. Minneapolis, MN.
- Figa, E. and P. Tarau. 2003. Story traces and projections: Exploring the patterns of storytelling. *First International Conference on Technologies for Interactive Digital Storytelling and Entertainment (TIDSE 2003)*, Darmstadt, Germany.
- Gebhard, P., M. Kipp, M. Klesen, and T. Rist. 2003. Authoring scenes for adaptive, interactive performances. *The Second International Conference on Autonomous Agents and Multiagent Systems (AAMAS-03)*, pp. 725–732. Melbourne, Australia.
- Gervas, P., B. Diaz-Agudo, F. Peinado, and R. Hervas. 2005. Story plot generation based on CBR. *Journal of Knowledge Based Systems* 18(2–3):235–242.
- Geurts, J., S. Bocconi, J. van Ossenbruggen, and L. Hardman. 2003. Towards ontology-driven discourse: From semantic graphs to multimedia presentations. In: *Proceedings of the International Semantic Web Conference 2003*, pp. 597–612. Sanibel Island, FL.
- Global Document Annotation (GDA) Initiative. <http://www.i-content.org/GDA/>. Last accessed 2009/06/22.
- Graesser, A. and L. Clark. 1985. *Structures and Procedures of Implicit Knowledge*. Norwood, NJ: Ablex Publishing Corporation.
- ISIRST Project website. <http://www.ninfort.com>. Last accessed 2009/06/22.
- Java Media Framework (JMF). <http://java.sun.com/javase/technologies/desktop/media/jmf/>. Last accessed 2009/06/22.
- JLayer MP3 Library. <http://www.javazoom.net/javalayer/sources.html>. Last accessed 2009/06/22.
- Kaon2 Reasoning Engine. <http://kaon2.semanticweb.org/>. Last accessed 2009/06/22.
- Lang, P. J. 1984. *Cognition in Emotion: Concept and Action. Emotions, Cognition, and Behavior*, pp. 192–226. New York: Cambridge University Press.
- Levenson, R. W. 1988. Emotion and the autonomic nervous system: A prospectus for research on autonomic specificity. In: *Social Psychophysiology and Emotion: Theory and Clinical Applications*, pp. 17–42, New York: John Wiley and Sons.
- Little, S. and J. Geurts, J. Hunter. 2002. Dynamic generation of intelligent multimedia presentations through semantic inferencing. In: *Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries*, pp. 158–189. Rome, Italy.
- Loquendo TTS Demo. [http://www.loquendo.com/en/demos/demo\\_tts.htm](http://www.loquendo.com/en/demos/demo_tts.htm). Last accessed 2009/06/26.

- Magerko, B. and J. E. Laird. 2003. Building an interactive drama architecture. In: *First International Conference on Technologies for Interactive Digital Storytelling and Entertainment (TIDSE-2003)*, pp. 226–237.
- Magerko, B. and J. E. Laird. 2004. Mediating the tension between plot and interaction. In: *AAAI Workshop Series: Challenges in Game Artificial Intelligence*, pp. 108–112. San Jose, CA.
- Mann, W. and S. Thompson. 1987. *Rhetorical Structure Theory: A Theory of Text Organization*, Marina Del Rey, CA: Information Sciences Institute.
- Mateas, M. and A. Stern. 2003. Façade: An experiment in building a fully-realized interactive drama. *Game Developers Conference, Game Design Track*, San Jose, CA.
- Mott, B. and J. Lester. 2006. U-DIRECTOR: A decision-theoretic narrative planning architecture for storytelling environments. In: *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*, pp. 977–984.
- Nakasone, A. and M. Ishizuka. 2006. SRST: A storytelling model using rhetorical relations. In: *Third International Conference on Technologies for Interactive Digital Storytelling and Entertainment, TIDSE 2006*. Springer LNCS 4326:127–138.
- Nakasone, A. and M. Ishizuka. 2007. ISRST: An interest based storytelling model using rhetorical relations. *Edutainment 2007*. Springer LNCS 4469:324–335.
- Nakasone, A., H. Prendinger, and M. Ishizuka. 2005. Emotion recognition from electromyography and skin conductance. *The Fifth International Workshop on Biosignal Interpretation (BSI-05)*, pp. 219–222. Tokyo, Japan.
- Protégé Ontology Editor and Knowledge Acquisition System. <http://protege.stanford.edu>. Last accessed 2009/06/26.
- Riedl, M. O., C. J. Saretto, and R. M. Young. 2003. Managing interaction between users and agents in a multi-agent storytelling environment. *The Second International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS)*, pp. 741–748. Melbourne, Australia.
- Rober, N., C. Huber, K. Hartmann, M. Feustel, M. Masuch. 2006. Interactive audiobooks: Combining narratives with game elements. In: *Third International Conference on Technologies for Interactive Digital Storytelling and Entertainment, TIDSE 2006*. Springer LNCS 4326:358–369.
- Sgouros, N. M., G. Papakonstantinou, and P. Tsanakas. 1997. Dynamic dramatization of multimedia story presentations. *ACM International Conference on Intelligent User Interfaces (IUI-97)*, pp. 87–94. Orlando, FL.
- Sheldon, L. 2004. *Character Development and Storytelling for Games*. Course Technology PTR.
- Silva, A., R. Guilherme, and A. Paiva. 2003. Tell me that bit again ... Bringing interactivity to a virtual storyteller. In: *Proceedings of the 2nd International Conference on Virtual Storytelling*, pp. 146–154. Toulouse, France.
- Sobral, D., I. Machado, and A. Paiva. 2003. Managing authorship in plot conduction. In: *Virtual Storytelling: Using Virtual Reality Technologies for Storytelling*, pp. 57–64. Strasbourg, France: Springer.
- Sumi, K., and K. Tanaka. 2005. Automatic conversion from E-content into animated storytelling. *Entertainment Computing – ICEC 2005* pp. 24–35.
- Swartout, W., R. Hill, J. Gratch, W. L. Johnson, C. Kyriakakis, K. Labore, R. Lindheim, S. Marsella, D. Miraglia, B. Moore, J. Morie, J. Rickel, M. Thiebaut, L. Tuch, and R. Whitney. 2001. Toward the Holodeck: Integrating graphics, sound, character and story. In: *Proceedings of the 5th International Conference on Autonomous Agents*, pp. 409–416. Montreal, Canada.
- Synchronized Multimedia Integration Language (SMIL). <http://www.w3.org/TR/REC-smil/>. Last accessed 2009/06/22.
- Szilas, N. 2003. IDtension: A narrative engine for Interactive Drama. In: *1st International Conference on Technologies for Interactive Digital Storytelling and Entertainment (TIDSE 2003)*, pp. 187–203. Darmstadt, Germany.
- Tan, E. 1996. *Emotion and the Structure of Narrative Film*, Mahwah, NJ: Lawrence Erlbaum Associates.
- Web Ontology Language (OWL). <http://www.w3.org/2004/OWL/>. Last accessed 2009/06/22.
- Zillmann, D. 1988. Mood management: Using entertainment to full advantage. In: *Communication, Social Cognition and Affect*, pp. 147–171. Hillsdale, NJ: Lawrence Erlbaum Associates.