A Description Logic Ontology for Fairy Tale Generation

Federico Peinado, Pablo Gervás, Belén Díaz-Agudo

Dep. Sistemas Informáticos y Programación
Universidad Complutense de Madrid, Spain
fpeinado@fdi.ucm.es, {belend,pgervas}@sip.ucm.es
http://gaia.sip.ucm.es

Abstract

The combination of resources like Ontologies and an inference formalism such as Description Logics has proved very useful for generating semantically correct texts. However the possibilities of applying such combinations to obtain results in practical situations is restricted by the availability of ontological resources for the domains under consideration. This paper presents work on the development of an OWL ontology based on Propp's *Morphology of the Folk Tale* oriented towards automatic story generation. The ontology is designed so that it allows measurement of the semantical distance between narrative functions. We explain how to use this resource to generate creative and meaningful stories.

1. Introduction

Certain properties of structured domains, like the syntax of formal poetry, make them particularly suitable to modeling in terms that allow automatic generation of elements belonging to that domain. This may be achieved by applying formal techniques of knowledge representation like Ontologies and Description Logics (DL). We have found ontologies and description logics a very powerful combination as a resource for generating linguistically creative correct texts (Díaz-Agudo et al., 2002). However the possibilities of applying such combinations to obtain results in practical situations is restricted by the availability of ontological resources for the domains under consideration. This paper presents work on the development of an OWL ontology oriented towards automatic story generation.

Automatic construction of story plots has always been a longed-for utopian dream in the entertainment industry, specially in the more commercial genres that are fuelled by a large number of story plots with only a medium threshold on plot quality, such as TV series or story-based video games.

The work of russian formalist Vladimir Propp on the morphology of folk tales (Propp, 1968) provides a formalism to describe the composition of folk tales as a structured domain. In this paper we describe the conversion of Propp's morphology into OWL description logic format (Bechhofer et al., 2004). The choice of OWL as representation language provides the additional advantage, that it is designed to work with inference engines like RACER (Haarslev and Möller, 2003), and that it is easily connected with Protégé (Gennari et al., 2002). This constitutes an extremely powerfull development environment, well suited for exploring linguistic creativity, and we hope to use it for exploring issues of story generation.

The resulting resource is employed as underlying representations for a Knowledge Intensive Case-Based Reasoning (KI-CBR) approach to the problem of generating story plots from a case base of Propp functions. A CBR process is defined to generate plots from a user query specifying an initial setting for the story, using the ontology to measure the semantical distance between words and structures taking part in the texts.

2. Theories and Implementations of Plot Generation

The automatic generation of stories requires some representation for plot structure and how it is built up from primitives, a computational solution to generating stories from a given input, and the choices of some format for presenting the resulting plots that is easy to understand and to generate.

2.1. General Theories on Plot Generation

In the first chapters of Seymour Chatman's *Story and Discourse* (Chatman, 1986) there is a review of various classical theories about narrative structures. Janet Murray shows another short review in the seventh chapter of her popular book *Hamlet on the Holodeck* (Murray, 1997). For example, she mentions Joseph Campbell's morphology of the mythic "hero (Campbell, 1972).

Our work is based on the work of Vladimir Propp (Propp, 1968), because it is easy to understand and translate into a machine-processable representation (the author brings us his own formal naming system). However there are other theories (Lakoff, 1972; Barthes, 1966) that propose more complex grammars and "deeper representations.

Propp's original goal was to derive a morphological method of classifying tales about magic, based on the arrangements of 31 "functions". The result of Propp's work is a description of the folk tales according to their constituent parts, the relationships between those parts, and the relations of those parts with the whole. Propp's work has been used as a basis for a good number of attempts to model computationally the construction of stories.

The main idea is that folk tales are made up of ingredients that change from one tale to another, and ingredients that do not change. According to Propp, what changes are the names - and certain attributes - of the characters, whereas their actions remain the same. These actions that act as constants in the morphology of folk tales he defines as *functions*.

For example, some Propp functions are: Villainy, Departure, Acquisition of a Magical Agent, Guidance, Testing of the hero, etc. There are some restrictions on the choice of functions that one can use in a given folk tale, given by implicit dependencies between functions: for instance, to be able to apply the Interdicion Violated function, the hero must have received an order (Interdiction function).

The Proppian fairy tale Markup Language (PftML) (Malec, 2004) is an XML application developed by University of Pittsburgh's researchers based on Propp's work. PftML utilizes a Document Type Definition (DTD) to create a formal model of the structure of Russian magic tale narrative and to help standardize the tags throughout a corpus when analyzing it. As a test corpus, they have used a subset of the same Russian language corpus from which Propp drew, since it allows for an empirical test of the conclusions of Propp's initial analysis against the original data.

We have used PftML, together with Propp's original work, as the basic sources for building the ontology that underlies our system.

2.2. Computer Models for Narrative

There have been various attempts in the literature to obtain a computational model of story generation. Important efforts along these lines are presented in (Meehan, 1981; Rumelhart, 1975; Lang, 1997; Callaway and Lester, 2002).

Fairclough and Cunningham (Fairclough and Cunningham, 2003) implement an interactive multiplayer story engine that operates over a way of describing stories based on Propp's work, and applies case-based planning and constraint satisfaction to control the characters and make them follow a coherent plot.

Of particular interest is their definition of a plot as a series of character functions and a series of complication-resolution event pairs, where a complication occurs whenever a character performs a function that alters the situation of the hero. A case based reasoning solution is used for storyline representation and adaptation. They use 80 cases extracted from 44 multi-move story scripts given by Propp. These scripts are defined as lists of character functions. There are stories composed of one, two or more moves. A case is a move, seen asa story template, to be filled in by a constraint satisfaction system that chooses which characters perform the functions - *casting*.

2.3. Template-based Natural Language Generation

The natural format for presenting a plot to users is to describe it - or rather narrate it - in natural language. Obtaining a high quality natural language text for a story is itself a subject of research even if the plot is taken as given (Callaway and Lester, 2002). This paper is concerned strictly with the process of generating valid plots, and only the simplest sketch of a natural language rendition is attempted as means of comfortably presenting the results. This is achieved by means of natural language generation (NLG) based on templates. The conventionalized patterns that make up common texts are encapsulated as *schemas* (McKeown, 1982), template programs which produce text plans. The basic resource required to apply this type of solution is a set of templates, obtained from the analysis of a corpus of example texts.

As in template-based NLG, Case-Based Reasoning (CBR) relies heavily on reusing previous solutions to solve new probles, drawing on a *case base* of existing problem-solution pairs enconded as *cases*. In (Díaz-Agudo et al., 2002) poetry generation is chosen as an example of the use of the COLIBRI (*Cases and Ontology Libraries Integration for Building Reasoning Infrastructures*) system. COLIBRI assists during the design of KI-CBR systems that combine cases with various knowledge types and reasoning methods. It is based on CBROnto (Díaz-Agudo and González-Calero, 2000; Díaz-Agudo and González Calero, 2001; Díaz-Agudo and González Calero, 2003), an ontology that incorporates reusable CBR knowledge and serves as a domain-independent framework to develop CBR systems based on generic components like domain ontologies and Problem Solving Methods (PSMs).

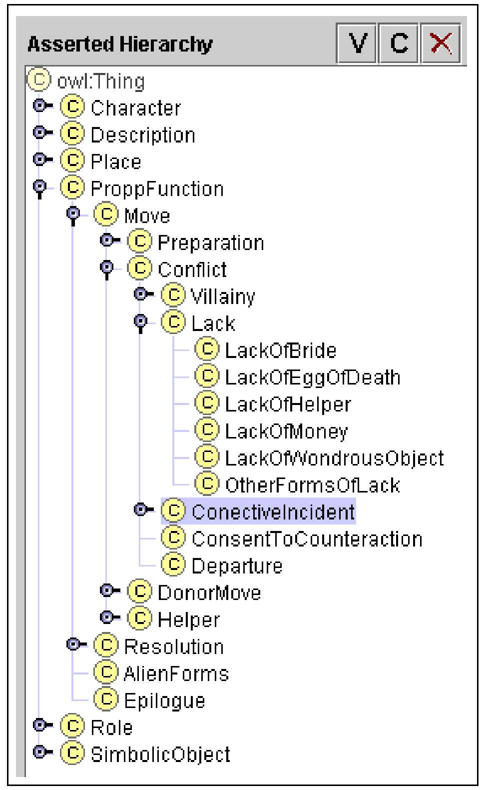

Figure 1: Function sub-hierarchy in the ontology as modelled in Protégé.

3. A DL Ontology for Fairy Tale Generation

Knowledge representation in our system is based on an ontology which holds the various concepts that are relevant to story generation. This initial ontology is subject to later extensions, and no claim is made with respect to its ability to cover all the concepts that may be necessary for our endeavour.

The ontology has been designed to include various concepts that are relevant to story generation. Propp's character functions are used as basic recurrent units of a plot. In order to be able to use them computationally, they have been translated into an ontology that gives semantic coherence and structure to our cases. A view of the top of the function hierarchy is given in figure 1.

Roles	Place	Character	Description	Simbolic Object
Agent Donor FalseHero Hero Prisoner Villain	City Country Dwelling	AnimatedObject Animal Human	Family description Human description Place description	Ring Towel

Table 1: Summary of additional subconcepts of the ontology

We have implemented this ontology using the last release of the Protégé ontology editor (Gennari et al., 2002), capable of managing ontologies in OWL(Bechhofer et al., 2004).

Although the functions of the *dramatis personae* are the basic components, we also have other elements. For instance, conjunctive elements, motivations, forms of appearance of the dramatis personae (the flying arrival of a dragon, the meeting with a witch), and the attributive elements or accessories (a witch's hut or her clay leg) (Propp, 1968).

This additional ontology provides the background knowledge required by the system, as well as the respective information about characters, places and objects of our world. This is used to measure the semantical distance between similar cases or situations, and mantaining a independent story structure from the simulated world. The domain knowledge of our application is the classic might-and-magic world with magicians, warriors, thieves, princesses, etc. The current version of the ontology contains a number of basic subconcepts to cover this additional domain knowledge that needs to be referred from within the represented function. Examples of these subconcepts are listed in table 1, including the character's roles proposed by Propp.

3.1. Propp's Terminology

In our approach, Propp's *character functions* act as high level elements that coordinate the structure of discourse. Each function has constraints that a character that is to perform it must satisfy. A view of the top of the function hierarchy is given in Figure 1.

The contents of a function are the answers to the Wh-questions: what (the symbolic object), when, where (the place), who (who are the characters of the function) and why.

Morphologically, a tale is a whole that may be composed of *moves*. A move is a type of development proceeding from villainy or a lack, through intermediary functions to marriage, or to other functions employed as a *denouement* (ending). Terminal functions are at times a reward, a gain or in general the liquidation of a misfortune, an escape from pursuit, etc. (Propp, 1968).

One tale may be composed of several moves that are related between them. One move may directly follow another, but they may also interweave; a development which has begun pauses, and a new move is inserted.

We represent tales and their composing moves using structured descriptions. A tale is related with an ordered sequence of complete moves. We represent the temporal sequence between these moves using the CBROnto temporal relations.

3.2. Background Knowledge

The ontology includes a significant amount of background knowledge needed for the successful application of the rest of its structure to the problem in hand.

Certain *locations* can be significant to the way a story develops (outdoors, indoors, country, city, lake, forest ...), and any sort of substitution during adaptation must take this into account. Our ontology must have the ability to classify such locations.

The roles in the story must be filled by *characters*. Each character is defined by a set of relationships with other characters, objects in his possession, location... These characters are one of the elements that the user can choose to customize a story.

The *descriptions* are represented in the ontology in such a way that their relations with the relevant concepts are modelled explicitly. This ensures that the inference mechanisms available can be employed to select the correct descriptions during the template-based NLG process which obtains a textual rendition of the plot.

The *properties or attributes of the characters* are the totality of all their external qualities: their age, sex, status, external appearance, peculiarities of this appearance,... These attributes provide the tale with its brilliance, charm and beauty. However, one character in a tale is easily replaced by another (permutability law) (Propp, 1968).

3.3. The Case Base

The case base is built up of texts from the domain of fairy tales, analyzed and annotated according to Propp's morphology. A selection of stories from the original set of the Afanasiev compilation originally used by Propp are taken as sources to generate our initial case base.

We use a structural CBR approach that relies on cases that are described with attributes and values that are predefined, and structured in an object-oriented manner. This structural CBR approach is useful in domains (like the one we are considering) where additional knowledge, beside cases, must be used in order to produce good results. The domain ontology insures that new cases are of high quality (regarding the ontology commitments) and the maintenance effort is low.

Within the defined case structures we represent the plots of the fairy tales. Besides this structural representation of the cases we also associate a textual representation to each case that can be used to generate texts from the plot descriptions (see Section 4.2.).

Cases are built based on CBROnto case representation structure (Díaz-Agudo and González Calero, 2003) using the vocabulary from the domain ontology. The semantic constraints between scene transitions are loosely based on the ordering and co-occurrence constraints established between Proppian functions.

CBROnto provides a *primitive* concept CASE. System designers will have to define instances of different CASE subconcepts to represent any new types of cases. There are different level of abstraction that allow the description of cases that are part of other cases.

In our application each case represents a complete tale that is typically composed of one or more interrelated moves (that are also cases). For representational purposes, relation between moves are basically of two types: *temporal* relations (before, after, during, starts-before, ends-before, ...) or *dependencies* (meaning that a change in one of them strongly affects the other) like *place-dependency*, *character-dependency* and *description-dependency* (Díaz-Agudo and González Calero, 2001).

DLs allows representing hierarchies between relations (see Figures 2 and 3), which allows easy definition of reasoning methods (using the top level relation) that are applicable (and reusable) with all the sub-relations.

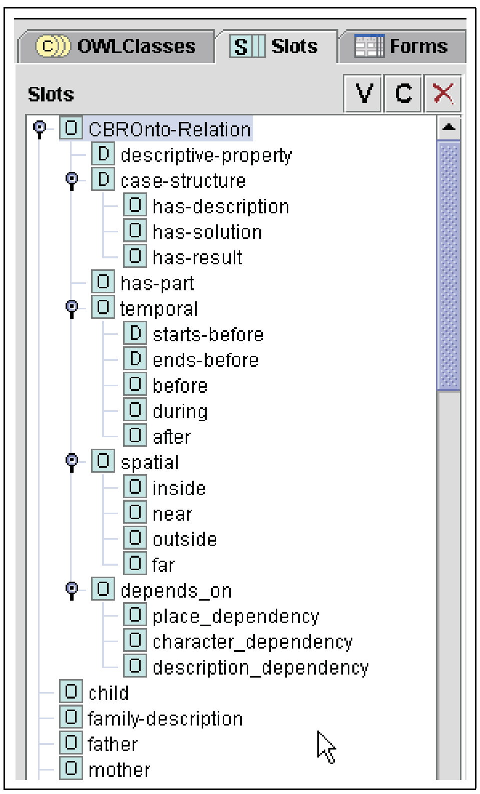

Figure 2: CBROnto relation hierarchy in Protege

As an example of the type of stories that are being considered, the following outline of one of the tales that Propp analyzes is given below [1]. The main events of the plot are

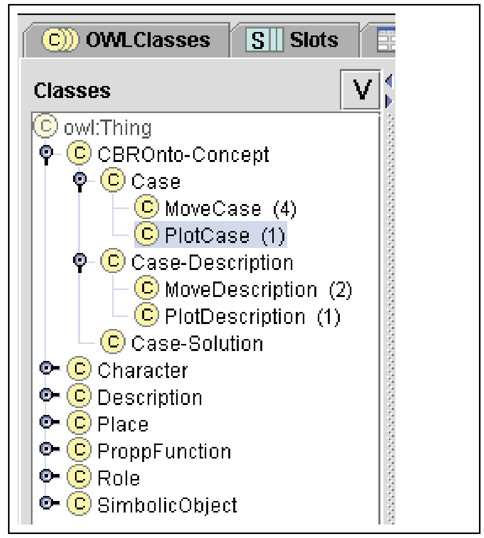

Figure 3: CBROnto concept hierarchy in Protege

described in terms of character functions (in bold) :

> *The Swan Geese (113 of Afanasiev Collection).* **Initial situation** (a girl and her small brother). **Interdiction** (not to go outside), **Interdiction violated**, **Kidnapping** (swan geese take the boy to Babayaga's lair), **Competition** (girl faces Babayaga), **Victory**, **Release from captivity**, **Test of hero** (swan geese pursue the children), **Sustained ordeal** (children evade swan geese), **Return**.

4. Ontologies and Case Base Reasoning in Plot Generation

The resources that are described in this paper are applied to the problem of generating story plots in two phases: an initial one that applies CBR to obtain a plot plan from the conceptual description of the desired story provided by the user, and a final phase that transforms the resulting plot plan into a textual rendition by means of template based NLG.

4.1. The First Stage: Description to Plot Plan

We use the descriptive representation of the tale plots with a CBR system, that retrieves and adapts these plots in several steps using the restrictions given in the query.

A query determines the components of the tale we want to build. For example, its characters, descriptive attributes, roles, places, and the Propp functions describing the actions involved in the tale. Although there are roles whose existence (a character that plays that role) is mandatory in every plot, like the hero and the villain, they are not required in the query as they can be reused from other plots (cases).

In a query the user describes: the tale characters, roles, places, attributes, the set of character functions that are to be involved in the story,and (optionally) which characters take part in each function.

This is done by selecting individual (predefined instances) from the ontology (see Figure 1) or creating new

[1] Complete text in: http://gaia.sip.ucm.es/people/fpeinado/swan-geese.html

ones (new instances of the types of characters or places given by the ontology). The knowledge in the ontology (and the associated reasoning processes) can help the user in this selection while maintaining the corresponding restrictions.

The system retrieves the most similar case to the query which constitutes a plot-unit template. The components of the retrieved case are substituted for information obtained from the context, i.e. the query, the ontology and other cases, during the adaptation process.

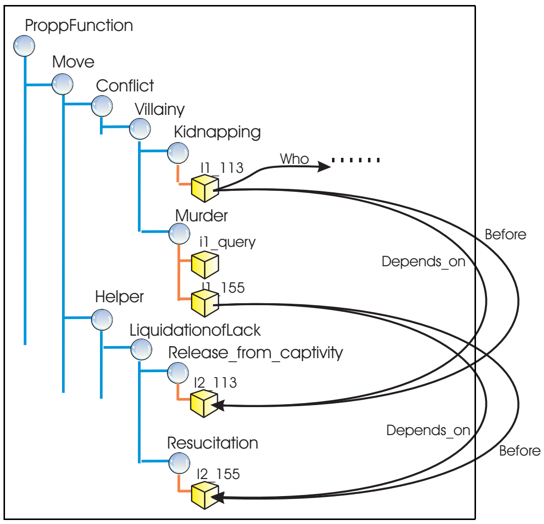

Figure 4: Substitution example

For instance, let us say we want a story about a *princess*, where **Murder** occurs, where an **Interdiction** is given and **Violated**, there is a **Competition**, and a **Test of the hero**. We can use that information to shape our query. The system retrieves the case story number 113, Swan-Geese (presented in the previous section).

Retrieval has occurred because the structure of this story satisfies straight away part of the conditions (interdiction, competition, test of hero) imposed by the query. No murder appears, but there is a *similar* element: a kidnapping. **Kidnapping** and **Murder** are similar because they are different types of villainies; so, they are represented as children of the same concept **Villainy** in the ontology.

The retrieval process provides the plot skeleton where the system makes certain substitutions. A basic and simple initial adaptation step is to substitute the characters given in the query into the template provided by the retrieved case. This is equivalent to Fairclough and Cunnigham's process of *casting*.

A more elaborate adaptation may be achieved by generating a solution as a mixture of the ingredients from various cases. During the adaptation of our *plot case*, we use additional retrieval steps (defining adequate queries) over the case base of *move cases* (that are part of the plot cases) to find appropriate substitutes maintaining the dependencies and temporal relations.

In our example, the system may suggest an adaptation where **Murder** is substituted for the **Kidnapping**. However, the **Kidnapping** in the retrieved case has *dependencies* with the **Release from captivity** that appears later on (which is a **Liquidation of lack** according to the ontology) (see Figure 4). To carry out a valid adaptation, the adaptation process is forced to define a query and retrieve cases in which **Murder** appears with a *similar* dependency (i.e. dependency with another **Liquidation of lack**).

The following case is retrieved (only a part of which is relevant to the issue):

> (*155 of Afanasiev Collection*). (...) **Absentation** of the hero (brother goes hunting), **Deception** of the villain (beautiful girl entices him), **Murder** (girl turns into lioness and devours him), (...) **Consent to counteraction** (other brother sets out), **Competition** (faces beautiful girl), **Victory** (kills lioness), **resuscitation** (revives brother), **Return**.

In this case there is a dependency between the **Murder** and the **Resuscitation**. The adaptation system can therefore substitute the kidnapping-release pair in the first retrieved case with the murder-resuscitation pair in the second, obtaining a better solution for the given query. Additional adaptations can be carried out to substitute the hero of the first case (the girl) or the prisoner (the boy) for the princess specified in the query. Besides, the swan-geese character in the retrieved case can be substituted for a similar element (for instance, another animal like the lioness that appears in the second retrieved case). The second part of *The Swan-Geese* story is not possible because of the lioness' death.

The resulting plot could be a story like this:

> *The Lioness (new fairy tale).* **Initial situation** (a knight and his beloved princess). **Interdiction** (not to go outside), **Interdiction violated**, **Murder** (a lioness devours her), **Competition** (knight faces the lioness), **Victory** (kills lioness), **Resuscitation** (revives the princess), **Return**.

4.2. The Second Stage: Plot Plan to Textual Sketch

A readable rendition of the plot plan is obtained by applying template-based natural language generation. The second stage takes as input a data structure satisfying the following constraints:

- The case that has been selected during retrieval, has been pruned or combined with other cases retrieved during adaptation and to make up a plot skeleton.
- The character functions, acting as templates for the basic units of the plot, have been filled in during adaptation with identifiers for the characters described in the query

A one-to-one correspondence can be established between character functions in the plot plan and sentence templates to be expected in the output and a simple stage of surface realization is applied to the plot plan. This stage converts the templates into strings formatted in accordance to the orthographic rules of English - sentence initial letters are capitalized, and sentences are ended with a colon.

The fact that we are using an ontology to represent concepts, and not a set of axioms encoding their meaning somehow restricts the degree of correctness that can be guaranteed by the substitution process. Any checking algorithm can only test for structural equivalence within the ontological taxonomy, and it cannot carry out proper inference over the meanings of concepts.

5. Conclusions

A major point of discussion that should be taken into account is whether Propp's formalism does constitute a generic description of story morphology. Without entering into that discussion here, it is still necessary to consider whether the procedure described in the paper enables the system to build new stories in a creative manner, or whether it simply allows reinstantiation of those in the original corpus with new elements. Unlike the uses of Proppian functions in other systems, our approach represents character functions with more granularity. This allows the establishment of relations between characters and attributes and the functions in which they appear. Using this facility, a coherent character set can be guaranteed throughout the story. Additionally, dependencies between character functions are modeled explicitly, so they can be checked and enforced during the process of plot generation without forcing the generated plots to be structurally equivalent to the retrieved cases.

The coverage of the ontology is an open issue dependent on whether one accepts Propp's set of character functions as complete. In the face of disagreement, the ontology is easy to extend, and, as mentioned before, it is not intended to be complete as it is. Under these conditions, the approach described in this paper may be extend to work in other domains.

Systems attempting to model linguistic creativity in the field of story generation would greatly benefit from incorporating semantic information in the form of a knowledge rich ontology such as the one described here. In future work we intend to address the specific problems of the natural language generation, involving the transition from plot plan to textual sketch, and to explore the possible interactions between the two stages.

6. Acknowledgements

The first author is supported by a FPI Predoctoral Grant from Complutense University of Madrid. The work was partially funded by the Spanish Committee of Science & Technology (TIC2002-01961).

7. References

Barthes, Roland, 1966. Introduction à lanalyse structurale des récits (in french). *Communications*, 8:1–27.

Bechhofer, Sean, Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L McGuinness, Peter F Patel-Schneider, and Andrea Stein, 2004. OWL web ontology language reference. http://www.w3.org/TR/2004/REC-owl-ref-20040210/.

Callaway, Charles B. and J. C Lester, 2002. Narrative prose generation. *Artificial Intelligence*, 139(2):213–252.

Campbell, Joseph, 1972. *Myths to live by*. New York: Viking Press.

Chatman, Seymour, 1986. *Story and Discourse : Narrative Structure in Fiction and Film*. Cornell University Press, 2nd edition.

Díaz-Agudo, Belén, Pablo Gervás, and Pedro González-Calero, 2002. Poetry generation in COLIBRI. In S Craw and A Preece (eds.), *ECCBR 2002, Advances in Case Based Reasoning 6th European Conference on Case Based Reasoning*. Aberdeen, Scotland: Lecture Notes in Artificial Intelligence. Springer.

Díaz-Agudo, Belén and Pedro Antonio González-Calero, 2000. An architecture for knowledge intensive CBR systems. In Enrico Blanzieri and Luigi Portinale (eds.), *Advances in Case-Based Reasoning*. New York: Springer-Verlag Berlin Heidelberg.

Díaz-Agudo, Belén and Pedro Antonio González Calero, 2001. A declarative similarity framework for knowledge intensive CBR. In *International Conference on Case-Based Reasoning*. Springer-Verlag.

Díaz-Agudo, Belén and Pedro Antonio González Calero, 2003. Knowledge intensive CBR through ontologies. *Expert Update*, 6(1).

Fairclough, Chris and Pádraig Cunningham, 2003. A multiplayer case based story engine. In *GAME-ON Conference*.

Gennari, J, M. A Musen, R. W Fergerson, W. E Grosso, M Crubézy, H Eriksson, N. F Noy, and S. W Tu, 2002. The evolution of Protégé: An environment for knowledge-based systems development. Technical report, Stanford University.

Haarslev, Volker and Ralf Möller, 2003. Racer user s guide and reference manual version 1.7.7. Technical report, Concordia University and University of Applied Sciences in Wedel.

Lakoff, George P, 1972. Structural complexity in fairy tales. *The Study of Man*, 1:128–150.

Lang, R. R, 1997. *A Formal Model for Simple Narratives*. Ph.D. thesis, Tulane University.

Malec, Scott A, 2004. Proppian structural analysis and XML modeling. http://clover.slavic.pitt.edu/ sam/propp/theory/propp.html.

McKeown, K. R, 1982. The text system for natural language generation: An overview. In *20th Annual Meeting of the ACL*. University of Toronto, Ontario, Canada.

Meehan, James R, 1981. Tale-spin and micro tale-spin. In Roger C. Schank and Christopher K. Riesbeck (eds.), *Inside computer understanding*. Hillsdale, NJ: Erlbaum Lawrence Erlbaum Associates.

Murray, Janet Horowitz, 1997. *Hamlet on the Holodeck. The Future of Narrative in Cyberspace*. Cambridge, MA: MIT Press.

Propp, Vladimir, 1968. *Morphology of the Folktale*. Austin: University of Texas Press.

Rumelhart, David E, 1975. Notes on a schema for stories. In Daniel G Bobrow and Allan Collins (eds.), *Representation and Understanding: Studies in Cognitive Science*. New York: Academic Press, Inc, pages 211–236.