

## Ontologies and Logic Reasoning as Tools in Humanities?

Amélie Zöllner-Weber <amelie\_dot\_zoellnerweber\_at\_gmail\_dot\_com>, Uni Digital, Bergen, Norway

### Abstract

In this article, we evaluate approaches using logic reasoning applied to an ontology for literary characters. The inference tool Racer and the programming language Prolog were tested and compared to see if they can serve as a computer assisted approach in this scenario.

Both offer options to draw inferences, but the usage requests a good comprehension of logics. Intuitive and individual queries are also difficult to produce with solely logic constructs. Furthermore, information in humanities is often vague, ambiguous, or even contradictory. Solving such problems by logic reasoning which is simply based on true or false conclusions will become challenging and might exceed the limits of logic reasoning.

At the moment, to the author's knowledge, only few such approaches, especially in literature studies, have been published. Existing approaches show promising results in modelling information in humanities. Therefore, further research should be directed to make ontologies and further approaches like logic reasoning even more popular and applicable in the humanities and literature studies.

## Introduction

Information seems to be essential for our lives. Humans get information everywhere, not only on the Internet, TV or radio channels and books, but also by talking to people or looking at road signs. We are constantly faced with problems and by solving these problems, our ability of using information is requested. But do we always find a solution just in time? And do we know when and where we can use information correctly? In daily life, we have to get information as quickly as possible. The Internet is a "mine" of information, but, like in a real mine, we need to light the darkness, to navigate by maps so that we are able to dig at the right point. The Internet is a giant source of information but often completely unstructured, leading many to the idea of structuring and grouping information presented on the Internet, the *Semantic Web*. Since the introduction of Semantic Web technologies and standards, like *RDF* and *OWL*, ontologies and other related methods have entered many research fields. In life science, they are already established, e.g. GeneOntology [Stevens et al. 2000]. Ontologies belong to approaches of information or knowledge representation. Now, this technology has been also introduced into several projects in the humanities, e.g. GOLD [Farrar et al. 2002], WordNet [Gomez 2004], TermNet [Lüngen 2007], FRBR [Renear et al 2006], and DISCOVERY [Smith 2007]. GOLD is an ontology for descriptive linguistics, e.g. syntax, morphology or linguistic data structures. WordNet is a lexical database of English that is also presented as an ontology. Thereby, TermNet, a terminological net for the domain of text technology, takes the concepts of WordNet into account. While these are linguistic ontologies, the FRBR ontology defines concepts of bibliographic cataloguing and indexing. Based on domain ontologies and other data bases, the DISCOVERY project links different philosophical projects together.

But does the obvious popularity of ontologies really point at the advantages? Or is the popularity only a trend in research?

There are some criticisms of ontologies [Veltman 2004], [Shirky 2005]. Veltman points out that the definition of meaning is often rather limited when using ontologies, especially in humanities. He argues that this definition is adequate for machine-to-machine transactions in, for example, business, but it does not address the needs of culture. There,

1

2

3

meaning comprises much more and needs to be handled by more suitable methods. In addition, Shirky criticizes the incomplete methods of classifying things. Summarized, he argues that categorization is always dependent on world views, which might not be accepted by other parties. Both authors hint at very important points, which should be discussed critically. But that goes beyond the scope of this contribution.

Nevertheless, the results of the approaches mentioned before show that ontologies and supporting technologies like logic reasoning are useful ways to express structured and hierarchical information in the humanities. In literature studies, the usage of ontologies is not very common. A reason for this might be that in this field, the contribution of an ontology to research topics is often rather difficult. Definitions of literary phenomena are unstable and unclear. Furthermore, in humanities and especially in literature studies, topics of research often cannot be measured like in life science [Zöllner-Weber & Apollon 2008]. As such it seems to be ever more challenging to introduce ontologies to these research areas, and, one might ask, which kinds of applications are offered and whether benefits might be drawn when applying such technologies.

Since their initiation in philosophy, ontologies, with their methods of subsumption related to semantics, have been expressed by logical expressions. Thereby, information about an ontology can be obtained using logic reasoning [Baader & Nutt 2003]. By formulating logic expressions, information that is often implicit or hidden in the ontology can be queried and conclusions can be drawn. This means that information about the structure and elements of ontologies can be retrieved, which can be of interest for human users, i.e. manual processing and automated processing, respectively. For example, even for human users, it might be difficult to get a detailed overview of relations between the elements in large ontologies without appropriate methods and utilities. Another interesting aspect might arise if ontologies from different sources are merged or mapped [Noy 2003, 18–9]. A merging or mapping may result in an enlarged description of the ontology domain.

Such a method is also of interest for schemes and models of topics in the humanities. Semantic relations that are essential in humanities can be expressed by creating an ontology. After storing information in an ontology, consecutive processing, like logic reasoning, creates further possibilities like consistency checks or support by merging data.

The aim of this work is to investigate probable applications for ontologies in the humanities, especially for literature studies. As mentioned, it is rather challenging to work with ontologies in humanities. In this contribution, an example ontology and its supporting tools, here focused on logic reasoning, are reviewed on three levels: a) the user's view, b) the methodological limits and c) limits of the tools with respect to literature studies.

In Section 2.1, an introduction to ontologies, especially the way they are used in Artificial Intelligence, is given, and their connection to logic reasoning is outlined. In Sections 2.2 and 2.3, an ontology for the description of literary characters and the application of logic reasoning to this ontology is presented [Zöllner-Weber 2006]. In Section 3, logics related to ontologies are introduced. Afterwards, the ontology for literary characters is used to test different logic-based methods; two tools, *Racer* and *Prolog*, are compared. *Racer* represents pure inference machines, whereas *Prolog* has been taken as an example for logic programming languages. In Section 4, the ontology and the applied logical reasoning tools regarding the aforementioned points are discussed. Finally, a conclusion and an outlook are given.

## Materials and Methods

In the next sections, an introduction to ontologies and especially to the ontology for literary characters is given.

### Introduction to Ontologies

Briefly, ontologies are used to provide organisation and retrieval of information semantically. Thereby, similarities between information provided by an ontology are searched by means of a semantic relation rather than by matching search strings or other similar measures. There are two comprehensions of the term of ontology: the pure formalistic approach of classifying objects as in the field of Artificial Intelligence and machine learning, and the more transcendental approach in the humanities and especially philosophy.

The term ontology, which originates from philosophy and contains the study of being or existence in general, was introduced by Aristotle [Philosophie-Lexikon 1991]. Aristotle was interested in describing the existence of things in the world. He asked fundamental questions like “What is existence?” On the one hand, he discussed human existence as it is still discussed in philosophy, and on the other hand, he developed a system (universals and particulars) of how the existence can be described, a formalism. In particular, this is based on descriptive logics. The system of these logics is contained in an ontology, which inherits formalistic principles, and therefore is a form of expressing logics. In the ancient world and maybe in Aristotle’s view, the different sciences did not exist separately. In today’s view, he mingled transcendental questions with pure formalism. Over the course of centuries, these two parts have been split into different sciences. In a simplified way, formalism and classification have been moved to natural science and mathematics while the transcendental core of Aristotle’s questions is discussed in philosophy and theology. By transferring the term of an ontology to the AI, it is more restricted and focused on modelling of concepts of the real world in computer systems. Gruber gives a definition: “An ontology is a formal, explicit specification of a shared conceptualisation” [Gruber 1993, 199]. In this approach, Gruber refers to groups of people with a common comprehension of the world. In that respect, the initial question of existence is no longer in focus but still has to be described. Therefore, the formalism relied upon in an ontology is used. An ontology in AI comprises hierarchically structured concepts, also called classes, of a part of the world (*a domain*). The representation should be produced in a machine-readable language. Noy et al. explain, “An ontology defines a common vocabulary for researchers who need to share information in a domain” [Noy et al. 2001, 1]. An ontology consists of a set of objects that are divided into classes, concepts, properties (also slots, or roles) and the restrictions of the roles [Noy et al. 2001, 3]. By relating main and more specialized classes to an ontology, a hierarchy can be created. Additionally, so-called instances of the classes represent individual objects of the selected domain. Although, the structure of an ontology is rather static, the included information can be queried and manipulated in several ways, e.g. using W3C standards like the *Web Ontology Language* (OWL) or *Resource Description Framework Schema* (RDFS) [Antoniou & Harmelen 2003]. OWL has been created to enlarge web sites with semantic information and to make the Internet usable as a structured information source [Ziegler 2004, 126].

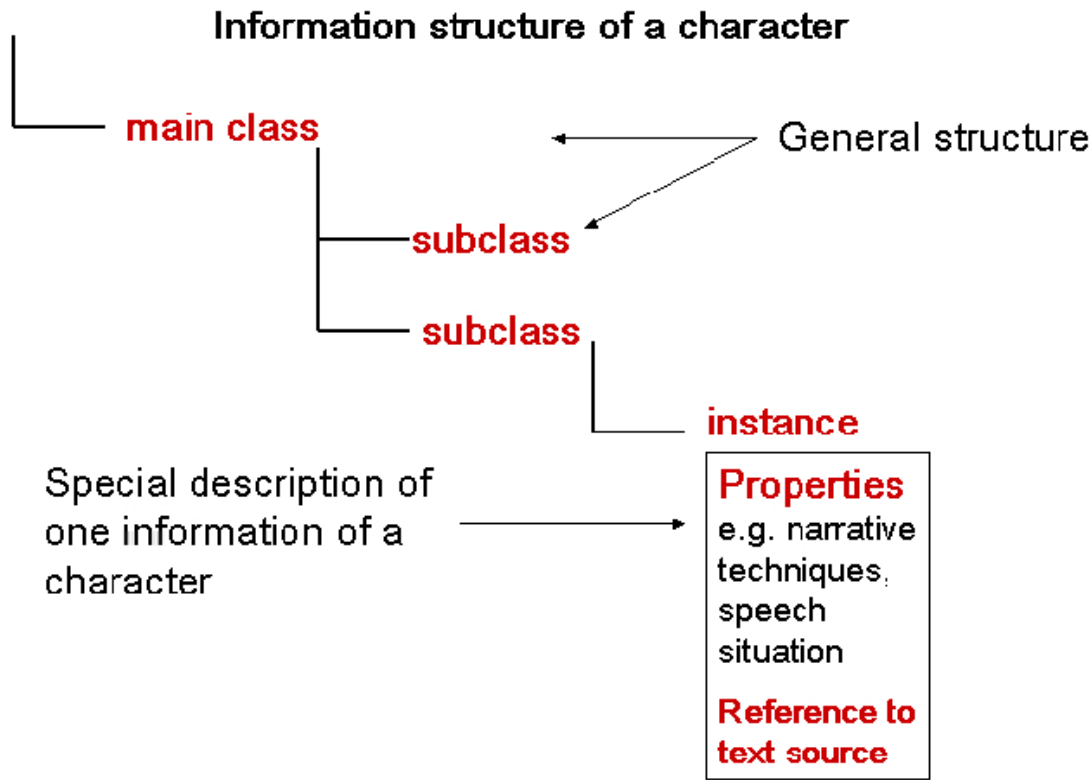
Discussion about ontologies, in this contribution, is focused on the formalistic method. Therefore, the discussion of existence, which has a long tradition in humanities, is not relevant here because an ontology is just seen as a very useful method to structure phenomena of literature studies. In the following, the ontology, criteria, and tools are presented.

## An Ontology for the Description of Literary Characters

To investigate ontologies and logic reasoning as tools in humanities, a concrete example is used. Here, the description of literary characters has been realised as an ontology.

Briefly, several theories of literary characters [Jannidis 2004], [Nieragden 1995] are combined to create a base of a formal description using an OWL ontology [Zöllner-Weber & Witt 2006]. Categories which describe features and actions of characters mentioned in stories had to be adapted and applied. It is aimed at representing the mental information structure of a reader, which (s)he has in mind when reading a book [Jannidis 2004, 185]. Categories describing general aspects of literary characters form the main classes of the ontology, e.g. inner and outer features, actions on other characters and objects. In this contribution, characters are regarded as individual objects, which are not linked to roles or archetypes [Propp 1968], [Greimas 1971]. Asking what a character means to a reader means not what kind of characters we have, but what general structure and attributes a character has. We regard a character as a complex cognitive entity in the reader’s mind, rather than emphasizing metaphors or archetypes. In this contribution, approaches that separate characters by roles, genres, etc., are faded out, because they might superimpose literary criticism on the daily phenomenon of reading literature. Reading literature and understanding characters does not necessarily require that everyone be a scholar in literature studies. It is of interest how a normal person as well as a scholar describes a character. By using a hierarchy, individual descriptions of characters should get a common ground to comparing these descriptions. By using sub classes, categories can subsume features of special characters or groups of characters. In addition, so-called instances of the classes represent individual and explicit objects of the domain of literary characters. Here, direct information about a character given in a text is assigned to an instance. Together with the information of the

class hierarchy and other instances, a single mental representation of a character is modelled (cf. Figure 1). In this approach, individual description, the pre-step of interpretation, is focused. Finding the same patterns revealed in different descriptions might be regarded as a common sense. By analysing these, similarities based on the cultural background of readers or of writing and reading traditions might be found. A more detailed description of this ontology is given in [Zöllner-Weber 2006], [Zöllner-Weber & Witt 2006].



**Figure 1.** Scheme of the representation structure of a character in the ontology. The scheme contains the distribution of the classes, instances, properties, and information about a character. Thereby, the main classes represent aspects that all kinds of characters have in common. The sub classes represent special features and actions for specific groups of characters. The instances represent pieces of information about a character. The properties contain additional information. Together, the classes, instances and properties form a mental representation of a character.

## Applications using the Ontology for Literary Characters

Having explained the concrete implementation of the ontology of literary characters, we focus this section on logic reasoning, which is a central principle when dealing with ontologies. 15

In general, conclusions drawn from logic reasoning can be inferred from given information. This means implicit information is derived from explicit information. In formal logic, given information is called assumption; the operation is called conclusion. The methods induction [Charniak & McDermott 1985, 22] and deduction [Charniak & McDermott 1985, 14] are subsumed under the term of logic reasoning. Induction means inferring from special concepts to general concepts whereas deduction constitutes the opposite process from general concepts to special ones. Here, the conclusions can only receive the values true or false. The assumptions of the induction have to be true so that a conclusion can be constituted as true as well. The conclusion is called an inference. 16

Logic reasoning is one possible application for ontologies. It is probably helpful (i) to check consistency during ontology development, (ii) to enable semi-automatic merging of (domain) ontologies as well as (iii) to deduce hidden information contained in the ontology. These three tasks can be applied to all elements of ontologies, classes as well as instances. As Baader et al. mention, logic reasoning can fulfill different purposes in the phase of creating an ontology and in the phase of using it [Baader & Nutt 2003, 4], e.g. investigating the structure of categories/ concepts, or testing if every 17

object is used in the intended and not contradictory way. In different situations of a work process, logic reasoning can be used to avoid or to solve problems: If several persons build together an ontology, new included elements can be checked for inconsistency or redundant information can be detected.

Coming back to the mentioned mapping processes, one can state that for the ontology of literary characters, it might be useful, if two persons add categories with the same or similar meaning to the ontology, to produce suggestions for mapping. Applications like mapping or merging are useful for the development and usage of ontologies. But recently, to the author's knowledge, there have been no other ontologies in literature studies which deal with literary characters or narratology and which could be related to this ontology, i.e. by merging or mapping. To be successful, synonymy has to be taken into account and a relating mechanism has to be implemented, because processing of synonymy is not included directly in an ontology so far. Therefore, these tasks cannot be applied completely to the ontology for literary characters at the moment.

18

Especially in the humanities, where textual data is often crucial, applications can be used which combine ontologies with text annotations or information retrieval. Ontologies consist of information, ideas, and/or facts, which also occur or are repeated in (text) sources that are not included in an ontology. This might be especially interesting when dealing with ontology learning. For example, terms contained in both sources, in text annotations/web pages and an ontology [Baader & Nutt 2003, 4] can be queried for. But here arises the question of whether arranging data that both sources can match would be a redundant and time-consuming work. In addition, setting up more elaborate match processes could require a complex system including lexica or external rules [Puppe et al. 2000, 635–6], [Benjamins et al. 1999].

19

Nevertheless, logic reasoning is not only useful for such processes. Logic reasoning can also help during users' orientation period. Hierarchical relations can be highlighted so users can get a structured overview about all elements and their relations. Another aspect is obtaining information that is implicitly hidden in the data.

20

These examples show only some of the imaginable methods which might be performed by logic reasoning, but already they outline possibilities for an ontology in the humanities. It does not matter for which discipline ontologies are developed; they all have to pass a development, test, and deployment phase and thus, the methods of logic reasoning might be helpful.

21

However, mapping or merging is yet not possible because of the singularity of the ontology for literary characters. Such operations might become possible if further ontologies in this field will become available.

22

In addition, the human user should be focused when testing logic reasoning. In order to test logic based applications, several tasks are defined, - which are similar to the queries given in linguistic approaches [Lüngen 2007]:

23

Retrieval of

24

- **Individuals/instances that belong to a given class.** The results should give information about the relationship between special classes and their instances so that it is possible to obtain information about a character, which is disseminated in a text, e.g. that characters consist of special features whose information is expressed as instances
- **Individuals/instances that contain a given property (and maybe a given value).** The results should give information about the relationship between instances and special properties and to which class the instance belongs so that the apportionment of additional information can be focused, thereby it might be possible to examine which special kinds of properties only belong to some classes
- **Bottom up/top down.** The result should show the relationship of classes and its super classes, it should also show the arrangement of the features/actions of characters and the structure of the mental representations

## Testing Applications of Logic Reasoning

### OWL and its Relation to Description Logic

The ontology used in this contribution was implemented in *OWL Description Logic* [Antoniou & Harmelen 2003]. The standard OWL consists of three sub languages (OWL Light, OWL DL, OWL Full) created on different levels of information representation. Before going on, a short summary of the Description Logic whose ideas are included in OWL DL: OWL is often used in its sub language OWL DL, as it inherits concepts of a logic formalism called *Description Logic* (DL). In DL, concepts are grouped in a so-called *TBox* and individuals belong to an *ABox*. The structure of a TBox is represented as subsumption so that more general concepts contain special ones, which correspond to the concepts/classes of an ontology. Regarding the ontology for literary characters, all classes that have been defined (cf. Figure 2) are collected in the TBox. An ABox subsumes all instances of the modelled domain. This can be seen as a description of a domain. Coming back to the ontology for literary characters, this means that the ABox is composed of the instances contained in the ontology.

As Nardi and Brachman explain, these methods are usually a variant of first-order predicate calculus (cf. [Nardi & Brachman 2003, 2]). Therefore, it is possible to use the mechanisms of logic formalism like reasoning or drawing inferences. To operate on data that is formatted or transformed into Description Logic, inference algorithms have been developed. These algorithms are implemented in different programmes like FaCT++ [Tsarkov & Horrocks 2006], Pellet [Sirin et al. 2007], and Racer [Haarslev et al. 2004], which are DL reasoners, or Prolog <http://www.swi-prolog.org/> which is also able to query OWL ontologies.

## Racer and Prolog

In the next sections, the two already-mentioned tools, the inference machine Racer and the logic programming language Prolog, will be described.

### Introduction to Racer

Here, we investigate whether Racer (*Renamed ABox and Concept Expression Reasoner Professional*) is a good tool for drawing inferences on ontologies in the humanities. Racer was developed for approaches that are based on OWL/ DL and offers inference functions as well as consistency checks (cf. [Racer Systems 2005, 1]).

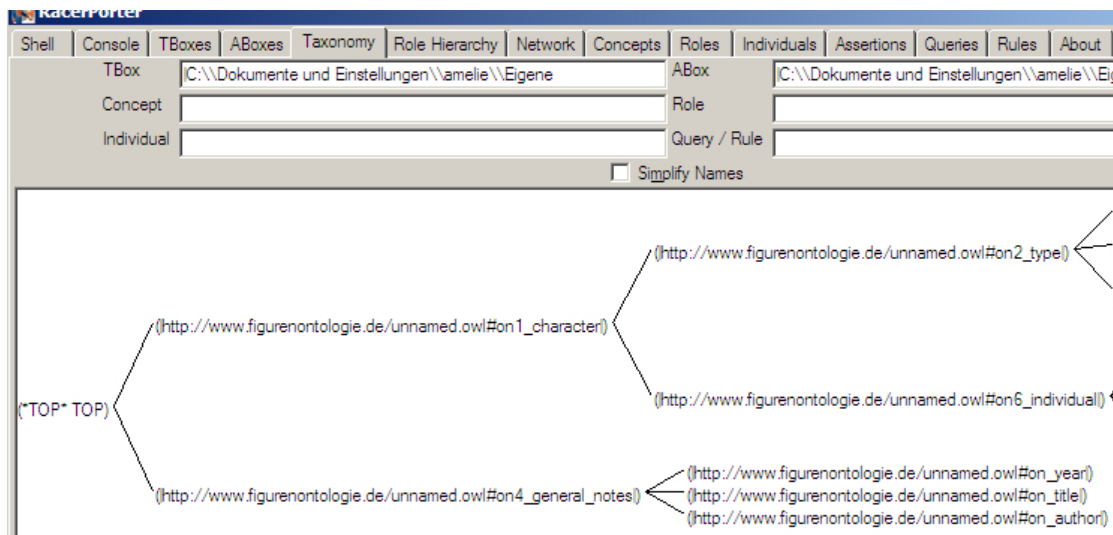


Figure 2. Extract of the program Racer. Specifically, an extract of the TBox, which shows several classes of the ontology for literary characters, is depicted.

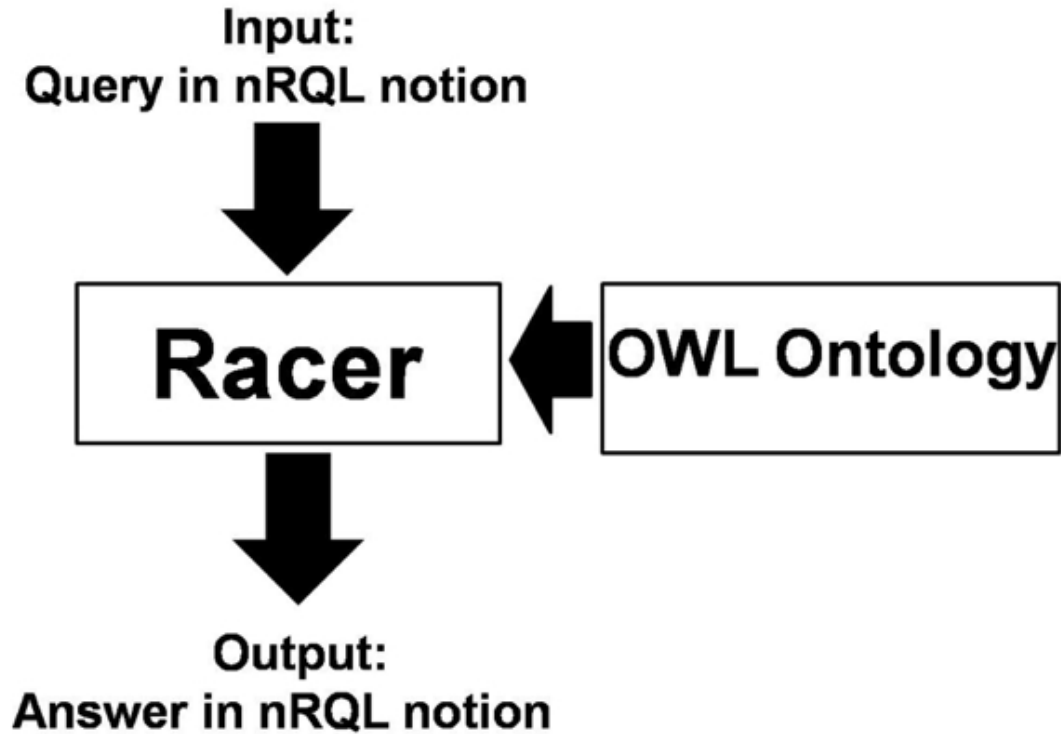
According to the concepts of an ABox and TBox in DL, the classes and individuals are presented and inferred separately (cf. Figure 3). To infer in OWL, Racer uses the query language *nRQL* [Racer Systems 2005, 87]. This means that the notion in the XML standard of OWL has to be translated to this query language. Queries in nRQL consist of a head and a body similar to predicate calculus [Haarslev et al. 2004].

The syntax for a TBox and ABox differs from each other. By an ABox query, for example possible individuals of a

special class can be retrieved. It is also possible to query for individuals with special properties. So, information nested in individuals and properties can be related to each other. In order to retrieve individuals with a data type property, so-called *head projection operators* have to be used. In a TBox, classes that do or do not have a given name can be also queried for.

Figure 3 shows how Racer is working in conjunction with an ontology. Except for the graphical notion of TBox and ABox, all queries in nRQL have to be inserted manually. The results are also shown in nRQL. The user has no graphical interface when querying in the ontology.

31



**Figure 3.** Scheme of interaction between Racer and an ontology. The querying of the ontology is realized via nRQL queries given to the Racer program.

Another helpful option is to use a special construct that can be used in combination with the classes that should be retrieved. This means that direct hierarchical relations between classes like sub or super relations can be retrieved. But one should note that Racer can only offer a result if the classes are in a hierarchical relation to each other.

32

In general, when querying an ontology by Racer, different results can be obtained. First, a Boolean is returned (true or false), i.e. an assumption about the ontology can be verified. Second, a literal value is returned, depicting certain information queried for.

33

All these efforts are reasonable when dealing with huge or (partly) unknown ontologies. Logic reasoning using Racer might help a human to orientate himself/herself and to work easily with such an ontology. Figure 4 shows the processing of possible queries of an ABox or TBox.

34

### 1. A user enters a query in nRQL:

**ABox:** (RETRIEVE(?X)  
(?X |http://www.figurenontologie.de/unnamed.owl#on4\_general\_notes|))

**TBox:** (TBOX-RETRIEVE(?X)  
(NEG (?X |http://www.figurenontologie.de/unnamed.owl  
#on43\_statement\_about\_the\_analised\_character|)))

### 2. System is processing query

### 3. Result is confirmed to user

**Figure 4.** Examples for nRQL queries. An ABox and a TBox are shown. The TBox query is an example for including negation.

## Introduction to Prolog

Another possibility for drawing inferences is to use a programming language that is specialized for logics, e.g. Prolog. It is based on first-order predicate calculus [Fisher 2006] and provides unification, backtracking, and tail recursion: operations that are especially useful for ontologies. Inferences can be drawn by using a parser that operates on a so-called *Prolog fact base*. This means that, similar to Racer and its query language, OWL ontologies have to be transferred to a notion in Prolog.

35

Data in Prolog consists of *facts* and *rules*. The facts can be considered a vocabulary on which Prolog works. Facts, called *predicates*, are defined with arguments included to describe the knowledge, and given to Prolog. A predicate is defined by giving a head and arguments (in brackets). By using rules, the fact base can be enlarged so that different relations and combinations between the modelled elements can be achieved.

36

Using Prolog as an inference tool for ontologies, the *SWI-Prolog* implementation offers several useful modules for manipulating RDFS and OWL data. In this approach, the RDFS module for SWI-Prolog has been used <http://www.swi-prolog.org/pldoc/package/semweb.html>. By loading an ontology, the module represents the data as triples in a Prolog fact base. The RDFS module can query for the relations between classes, properties, and individuals by using predefined predicates, e.g. relation between classes and instances. But one can also query for instances with a given property.

37

It is also possible to retrieve instances with a given literal, e.g. special, property value. For comparison reasons, the usage of literals like character names included in the ontology for literary characters is very useful. Thus, all the instances or pieces of information of two literary characters can be retrieved. But it might also be interesting to query for characters created by a special author. For example, a user wants to retrieve every instance with the name of a given author in its property. Information can be filtered. Hierarchical relations of classes, e.g. super-sub class relations can also be queried top down or vice versa. By combining the predicates of the module, a wide range of different queries

38



can be supported.

## Discussion

Having outlined the technical principles, in the following, we discuss the application of logic reasoning to an ontology in literature studies on three levels: a) the user's view, b) the methodological limits including ontologies, and c) limits of the tools especially for literature studies.

39

The task defined in Section 2.2 could be realised by queries formulated in nRQL and Prolog.

40

Table 1 in the appendix summaries all types of queries implemented and executed so far. The query in nRQL depicted in Figure 5, as an example, as well as in Table 2 in the appendix, demonstrates the possibilities of using logic reasoning.

41

Literally, it could be asked if a certain description of a character is featured by a class (instance-class relation), combination of content of the primary text and narrative/ additional information (instance-property relation), and the hierarchical structure of a character (top-down relation).

42

```
a) RETRIEVE(variable(TOLD-VALUE(|property storytitle|variable)))
(variable|class on43_statement_about_the_analised_character |))
(RETRIEVE(?X(TOLD-VALUE
(|http://www.figurenontologie.de/unnamed.owl#storytitle|?X)))
(?X|http://www.figurenontologie.de/unnamed.owl#on43_
statement_about_the_analised_character|))

b) ((variable||instance|)((:TOLD-VALUE|
propertystorytitle|variable))("content of the property"))
((?X|http://www.figurenontologie.de/unnamed.owl
#about_the_black_poodle|)
((:TOLD-VALUE|http://www.figurenontologie.de/
unamed.owl#storytitle|?X))("Faust Der Tragoedie Erster Teil"))
```

**Figure 5.** RDF Query (a) and its result (b). Show all instances that belong to the class `on43_statement_about_the_analised_character` and have the property `storytitle`. In b) only the first element returned (of a longer list) is given for brevity.

Racer and Prolog differ in their basic concept of logics: Prolog is based on predicate calculus whereas Racer uses the concept of description logic (ABox, TBox syntax). Both tools have been evaluated in this work. In comparing Prolog and Racer, a huge difference between them is that SWI-Prolog cannot only be used for drawing inferences — the parser can handle all kind of data represented as a Prolog fact base. In contrast, Racer is a solely inference machine.

43

Both have their own representation and syntax so that a user who is querying an ontology needs detailed knowledge on handling both programmes, i.e. to draw conclusion with these tools, the plain representation of the ontology in OWL, an XML standard, has to be transformed into the programmes' own representation in advance. Additionally, a possible user has to be familiar with the concepts of an ontology to formulate reasonable queries. This assumes certain knowledge in theory in ontologies and logic but also some technical "hands on" experience for daily use. It is questionable if these skills can be a requirement for a scholar with traditional education in literature studies without proper training. In summary, both programs are reasonable for inference tasks with ontologies in humanities. Both have advantages and disadvantages, especially formulating queries seem to be difficult. SWI-Prolog might be a choice since complete scripts can be executed. On the other hand, Racer provides a graphical representation of ontologies.

44

As mentioned, an ontology consists of a set of objects, concepts, and further entities, which are related to each other

45

[Noy et al. 1997, 53]. These objects are divided into classes, properties [Noy et al. 2001, 3]. Additionally, instances of the classes represent individual objects of the selected domain. By relating main classes and more specialised ones of an ontology, a hierarchy can be created. Therefore, only information in a hierarchical order can be modelled using an ontology. Similarities between concepts or classes commonly occurring when modelling certain information from literature cannot be expressed directly by an ontology. However, there exist certain structures within the OWL languages that allow for such relations. But these relations have to be modelled explicitly during the creation of the ontology. Concerning the description of literary characters as in this work, theories which tend to have a hierarchical structure for describing the characters as Nieragden [Nieragden 1995] should be included. Theories that are based more on network-like structures, however, should be modeled using topic maps or similar techniques, for example.

Besides this disadvantage, one has also to note that vague information can be handled by the ontology through the instantiation of a class. While the class describes the more general and common agreed on concept, the instance refers to a concrete example, also including possible individual interpretations. This kind of modelling has been used for the ontology of literary characters. 46

In principle, drawing inferences can retrieve implicit and hidden information. The structure of OWL DL allows logic and formal applications, but the scope of the logic-based queries is often restricted. Since the logic theorem is based on the closed world assumption, only information already included in a fact base can be returned. Mostly the implicit information that possibly exists in an ontology is anchored in the hierarchical relations between the classes and between classes and instances. Furthermore, the employed logical concept in description logics only can give back true or false conclusions (or the elements of the fact base that match a true result). However, when it is queried for vague information (not modelled via certain similarity constructs as explained before) this approach will fail since they cannot be expressed in logical terms [Berthold 2003]. Furthermore, such information is often ambiguous, or even contradictory [Zöllner-Weber & Apollon 2008]. Here, fuzzy logics or statistically methods might be superior. In computer science, many approaches combine logic reasoning with other methods of information retrieval to overcome the limits of logics. Logical queries are often embedded in systems, which contain further sources like lexica to extract more information e.g. ontology learning [Cimiano et al. 2006]. 47

More generally, when comparing literature studies and computer systems, one has to take into account that semantics in literature studies often do not match semantics in machine-to-machine learning [Veltman 2004]. Often, technical, e.g. machine-to-machine, semantics are more flat and not too complex, whereas in literature semantics could be highly complex. As an example in the present work, the description of the literary characters is dependent on the reader's interpretation. Therefore, the modelling of the ontology must allow for several interpretations side-by-side in the ontology (via instantiations). This is also covered in [McCarty 2005], where he points out that proper modelling in humanities is important and that the model has to be kept open or flexible, i.e. to be able to include the thinking itself. Furthermore, one has to note that the foundations i.e. the literary theories of character descriptions, on which the ontology is built on, mainly have hierarchical character and thus, are more advantageous for the proposed task. Nevertheless, also the structure of a hierarchy is critical. Shirky notes that hierarchies are usually built on a single world-view [Shirky 2005]. But mostly the weighting or the level of detail within different branches in the hierarchy is influenced by the modeller or by economical aspects. Therefore, the hierarchy might not be objective. This might not be a problem for machine-based processing; however, a user of this ontology might not agree on the presented hierarchy. This risks that the provided system will be abandoned by the user. 48

## Conclusion and Outlook

In this article, we evaluated approaches using logic reasoning applied to an ontology for literary characters. The inference tool Racer and the programming language Prolog were tested and compared to see if they can serve as a computer assisted approach in this scenario. 49

Both offer options to draw inferences, but the usage requests a good comprehension of logics. In addition, intuitive and individual queries are also difficult to produce with solely logic constructs. Furthermore, the hierarchy model of an ontology might often not be suitable for certain topics in literature studies. For example, strands of plots that evolve over 50

time cannot be represented by using a hierarchy. For such analysis, other approaches are more suitable (for example [Meister 1999]). One should also keep in mind that information in humanities is often vague, ambiguous, or even contradictory. An example of such a situation is given by the relations between language and the world in Ludwig Wittgenstein's philosophical work *Tractatus*. If trying to model these by an ontology (hierarchy), the author was faced by the fact that the hierarchy changes during the progress of Wittgenstein's approach [Zöllner-Weber & Pichler 2007]. However, a model represented by an ontology remains static. Solving such problems by logic reasoning which is simply based on true or false conclusions will become challenging and might exceed the limits of logic reasoning.

Nevertheless, the potential of logic reasoning for literature should be explored. It is helpful in the different phases of the creation and deployment of ontologies in the humanities and its sub disciplines like literature studies. For example, when different experts are working on the same ontology, its consistency can be checked and errors can be minimised in an early state of an ontology.

51

Probably, when dealing with such information and logic systems, Fuzzy Logic should be considered [Halpern 2005]. It offers a range of values that weight a conclusion rather than a binary decision as the logics described in this contribution. Also, combinations with other types of systems, like additional knowledge bases, might support current ontology approaches in humanities and literature studies.

52

At the moment, to the author's knowledge, only few such approaches, especially in literature studies, have been published. Therefore, a general judgement about the value of computer-aided approaches as outlined in this contribution could not be demonstrated to its limits. Existing approaches (for example [Farrar et al. 2002], [Lüngen 2007], [Zöllner-Weber & Witt 2006]) show promising results in modelling information in humanities. Therefore, further research should be directed to make ontologies and further approaches like logic reasoning even more popular and applicable in the humanities and literature studies.

53

## Appendix

Query for instance class relation:	<code>rdf(INSTANCE,'http://www.w3.org/1999/02/22-rdf-syntax-ns# type',CLASS), rdf(CLASS,'http://www.w3.org/1999/02/22-rdf-syntax-ns#type', 'http://www.w3.org/2002/07/owl#Class').</code>
Query for instances that belong to one class:	<code>rdf_has(X,Y,'http://www.figurenontologie.de/ #on27_action_and_behaviour').</code>
Query for top down relation:	<code>rdf_has(Y,Z,X),rdf_has(A,B,Y).</code>
Querying hierarchies:	<code>rdf_has(SUBCLASS,'http://www.w3.org/2000/01/rdf-schema#subClassOf',SUPERCLASS), rdf_has(SUPERCLASS,'http://www.w3.org/2000/01/rdf-schema#subClassOf',SUPERSUPER), rdf_has(SUPERSUPER,'http://www.w3.org/2000/01/rdf-schema#subClassOf',SUPER).</code>
Query for sibling classes:	<code>rdf_has(SUBCLASS1,'http://www.w3.org/2000/01/rdf-schema#subClassOf',SUPERCLASS), rdf_has(SUBCLASS2,'http://www.w3.org/2000/01/rdf- schema#subClassOf',SUPERCLASS),SUBCLASS1=SUBCLASS2.</code>

**Table 1.** Table 1 depicts several PROLOG queries formulated in this work for assessing the ontology of literary characters.

An ABox query:	(RETRIEVE(?X)(?X  http://www.figurenontologie.de/unnamed.owl# on4_general_notes ))
A TBox query with negation:	(TBOX-RETRIEVE(?X) (NEG (?X  http://www.figurenontologie.de/unnamed.owl#on43_statement_about_the_analised_character )))
Query with head projection operator:	(RETRIEVE(?X(TOLD-VALUE ( http://www.figurenontologie.de/unnamed.owl# storytitle ? X))) (?X http://www.figurenontologie.de/unnamed.owl# on43_statement_about_the_analised_character ))

**Table 2.** Table 2 depicts several Racer queries in nRQL formulated in this work for assessing the ontology of literary characters.

## Works Cited

- Antoniou & Harmelen 2003** Antoniou, G., and F. Van Harmelen. "Web Ontology Language: OWL". In S. Staab and R. Studer, *Handbook on Ontologies*. Berlin: Springer, 2003. pp. 67-92.
- Baader & Nutt 2003** Baader, F., and W. Nutt. "Basic Description Logics". In F. Baader D. Calvanese, D.L. McGuinness, P.F. Patel-Schneider and D. Nardi, *The Description Logic Handbook*. Cambridge: Cambridge University Press, 2003. pp. 43-95.
- Baader et al. 2003** Baader, F., I. Horrocks and U. Sattler. "Description Logics". In Steffen Staab and Rudi Studer, eds., *Handbook on Ontologies*. Berlin: Springer, 2003. pp. 3-28.
- Benjamins et al. 1999** Benjamins, V.R., D. Fensel, S. Decker and A. Gómez Pérez. "(KA)2: Building Ontologies for the Internet: A Mid-Term Report". *International Journal of Human-Computer Studies* 51 (1999), pp. 687-712.
- Berthold 2003** BertholdM. "Fuzzy Logic". In M. Berthold and D.J. Hand, eds., *Intelligent Data Analysis*. Berlin: Springer, 2003. pp. 321-350.
- Charniak & McDermott 1985** Charniak, E., and D. McDermott. *Introduction to Artificial Intelligence*. Reading MA: Addison-Wesley, 1985.
- Cimiano et al. 2006** Cimiano, Philipp, Johanna Völker and Rudi Studer. "Ontologies on Demand? A Description of the State-of-the-Art, Applications, Challenges and Trends for Ontology Learning from Text". *Information, Wissenschaft und Praxis* 57: 6-7 (2006), pp. 315-320.
- Farrar et al. 2002** FarrarS., Lewis W. D. and Langendoen D. T. "An Ontology for Linguistic Annotation". In *Semantic Web Meets Language Resources: Papers from the AAAI Workshop, Technical Report WS-02-16*. Menlo Park, CA: 2002. pp. 11-19.
- Fisher 2006** FisherJ. R. *Prolog Tutorial*. [http://www.csupomona.edu/~jrfisher/www/prolog\\_tutorial/contents.html](http://www.csupomona.edu/~jrfisher/www/prolog_tutorial/contents.html).
- Gomez 2004** Gomez, Fernando. "Grounding the Ontology on the Semantic Interpretation Algorithm". Presented at *GWC 2004. Proceedings of the Second Global WordNet Conference (2004)*, pp. 124-129.
- Greimas 1971** Greimas, A.J. *Strukturelle Semantik: Methodologische Untersuchungen*. Braunschweig: 1971.
- Gruber 1993** Gruber, T.R. "A translation approach to portable ontologies". *Knowledge Acquisition* 5: 2 (1993), pp. 199-220.
- Haarslev et al. 2004** Haarslev, V., R. Möller and M. Wessel. "Querying the Semantic Web with Racer + nRQL". Presented at *ADL '04. International Workshop on Applications of Description Logics (2004)*.
- Halpern 2005** Halpern, Joseph Y. *Reasoning about Uncertainty*. Cambridge, MA: MIT Press, 2005.
- Jannidis 2004** JannidisFotis. *Figur und Person - Beitrag zur historischen Narratologie*. Berlin: Walter de Gruyter, 2004.
- Lüngen 2007** Lüngen, H., and A. Storrer. "Domain Ontologies and Wordnets in OWL: Modeling Options". *Zeitschrift für Computerlinguistik und Sprachtechnologie* 22: 2 (2007), pp. 1-17.
- McCarty 2005** McCarty, Willard. *Humanities Computing*. New York and Basingstoke: Palgrave Macmillan, 2005.
- Meister 1999** MeisterJan Christoph. "Jenseits von Lagado. Literaturwissenschaftliches Programmieren am Beispiel des Kodierungsprogramms Move Parser 3.1". *Jahrbuch für Computerphilologie* 1 (1999), pp. 71-83.
- Nardi & Brachman 2003** Nardi, D., and R.J. Brachman. "An Introduction to Description Logics". In Franz Baader Diego Calvanese, Deborah McGuinness, Daniele Nardi and Peter Patel-Schneider, eds., *The Description Logic Handbook*.

Cambridge: Cambridge University PRes, 2003.

- Nieragden 1995** Nieragden G. *Figurendarstellung im Roman: Eine narratologische Systematik am Beispiel von David Lodges Changing Places und Ian McEwans The Child in Time*. Trier: Wissenschaftlicher Verlag Trier, 1995.
- Noy 2003** Noy, N.F. "Tools for Mapping and Merging Ontologies". In Steffen Staab and Rudi Studer, eds., *Handbook on Ontologies*. Berlin: Springer, 2003. pp. 365-384.
- Noy et al. 1997** Noy, N.F., and D.L. McGuinness. "The State of the Art in Ontology Design". *AI Magazine* 18: 3 (1997), pp. 53-74.
- Noy et al. 2001** Warning: Biblio formatting not applied. Noy N. F. McGuinness D. L. *Ontology Development 101: A Guide to Creating Your First ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880*. 2001. <http://www-ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.pdf>.
- Philosophie-Lexikon 1991** Hügli, A., and P. Lübcke, eds. *Philosophie-Lexikon*. Reinbek bei Hamburg: 1991.
- Propp 1968** Propp, Vladimir. *Morphology of the Folktale*. Austin and London: University of Texas Press, 1968.
- Puppe et al. 2000** Puppe F., Stoyan H. and Studer R. "Kapitel 15: Knowledge Engineering". In G. Görz C.-L. Rollinger and J. Schneeberger, eds., *Handbuch der Künstlichen Intelligenz*. München: Oldenbourg Verlag, 2000. pp. 599-641.
- Racer Systems 2005** Racer Systems. *RacerPro User's Guide*. <http://www.racer-systems.com/products/racerpro/users-guide-1-9.pdf>.
- Renear et al 2006** Renear, Allen, Yunseon Choi, Jin Ha Lee and Sara Schmidt. "Axiomatizing FRBR: An Exercise in the Formal Ontology of Cultural Objects". Presented at *Digital Humanities 2006* (2006). *Proceedings of Digital Humanities 2006* (2006), pp. 164-170. <http://www.jinhalee.com/wp-content/uploads/2014/01/DH2006.pdf>.
- Shirky 2005** Shirky, Clay. "Ontology is Overrated: Categories, Links, and Tags". Presented at *O'Reilly ETech conference* (2005).
- Sirin et al. 2007** Sirin, Evren, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur and Yarden Katz. "Pellet: A practical OWL-DL reasoner". *Journal of Web Semantics* 5: 2 (2007), pp. 51-53.
- Smith 2007** Smith, D.C.P. "Re-Discovering Wittgenstein". *Proceedings of the 30th International Wittgenstein Symposium* (2007), pp. 208-210.
- Stevens et al. 2000** Stevens, R., C.A. Goble and S. Bechhofer. "Ontology-Based Knowledge Representation for Bioinformatics". *Brief Bioinform* 1 (2000), pp. 398-414.
- Tsarkov & Horrocks 2006** Tsarkov, D., and I. Horrocks. "FaCT++ Description Logic Reasoner: System Description". Presented at *IJCAR 2006. Proceedings of the International Joint Conference on Automated Reasoning* (2006), pp. 292-297.
- Veltman 2004** Veltman, K.H. "Towards a Semantic Web for Culture". *Journal of Digital Information* 4: 4 (2004). <http://journals.tdl.org/jodi/article/view/113>.
- Ziegler 2004** Ziegler C. "Surfende Maschinen". In *Das XML-Kompendium*. Hannover: 2004. pp. 126-130.
- Zöllner-Weber & Apollon 2008** Zöllner-Weber Amélie, and Apollon Daniel. "The Challenge of Modelling Information and Data in the Humanities". *Proceedings of the International Conference on Explorations of New Spaces, Relations and Dynamics in Digital Media Ecologies* (2008).
- Zöllner-Weber & Pichler 2007** Zöllner-Weber Amélie, and Pichler Alois. "Utilizing OWL for Wittgenstein's Tractatus". *30th International Wittgenstein Symposium* (2007), pp. 248-250.
- Zöllner-Weber & Witt 2006** Zöllner-Weber Amélie, and Witt Andreas. "Ontology for a Formal Description of Literary Characters". Presented at *Digital Humanities 2006* (2006). *Proceedings of Digital Humanities 2006* (2006), pp. 350-352.
- Zöllner-Weber 2006** Zöllner-Weber, A. "Formale Repräsentation und Beschreibung von literarischen Figuren." In G. Braungart, P. Gendolla, and F. Jannidis (eds), *Jahrbuch für Computerphilologie* 7. mentis Verlag, Paderborn (2006), pp. 187-203.
- Zöllner-Weber 2007** Zöllner-Weber Amélie. "Noctua literaria - A System for a Formal Description of Literary Characters". In G. Rehm A. Witt and L. Lemnitzer, eds., *Data Structures for Linguistic Resources and Applications*. Tübingen: 2007. pp. 113-121.



This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.