Development and Alignment of a Domain-Specific Ontology for Question Answering

Shiyan Ou[1], Viktor Pekar[1], Constantin Orasan[1], Christian Spurk[2], Matteo Negri[3]
Research Group in Computational Linguistics, University of Wolverhampton, UK[1]
German Research Centre for Artificial Intelligence GmbH (DFKI), Germany[2]
Fondazione Bruno Kessler – FBK, Italy[3]
E-mail: {Shiyan.Ou; V.Pekar; C.Orasan}@wlv.ac.uk, Christian.Spurk@dfki.de, Negri@fbk.eu

Abstract

With the appearance of Semantic Web technologies, it becomes possible to develop novel, sophisticated question answering systems, where ontologies are usually used as the core knowledge component. In the EU-funded project, QALL-ME, a domain-specific ontology was developed and applied for question answering in the domain of tourism, along with the assistance of two upper ontologies for concept expansion and reasoning. This paper focuses on the development of the QALL-ME ontology in the tourism domain and its alignment with the upper ontologies – WordNet and SUMO. The design of the ontology is presented in the paper, and a semi-automatic alignment procedure is described with some alignment results given as well. Furthermore, the aligned ontology was used to semantically annotate original data obtained from the tourism web sites and natural language questions. The storage schema of the annotated data and the data access method for retrieving answers from the annotated data are also reported in the paper.

1. Introduction

With the appearance of Semantic Web technologies, it becomes possible to develop novel, sophisticated question answering systems, where ontologies are usually used as the core knowledge component. An ontology is a specification of a conceptualization (Gruber, 1993). There are two types of ontology: *domain ontology* and *upper ontology*. A domain ontology (or domain-specific ontology) represents a set of concepts which are specific to a domain and the relationships among these concepts. An upper ontology describes common concepts that are generally applicable across a wide range of domains. Several standardized upper ontologies are available for public use, such as WordNet (Fellbaum, 1998), SUMO (Niles & Pease, 2001), and OpenCyc (Lenat, 1995). A domain ontology is used in restricted-domain question answering systems to formalize domain knowledge and represent natural language questions and underlying unstructured information sources. Meanwhile, one or more upper ontologies are usually used as complements of the domain ontology to augment the available domain resources through semantic connections and definitions. This paper focuses on the development of a domain ontology and its alignment to the upper ontologies, carried out in the QALL-ME[1] project.

QALL-ME is an EU-funded project which aims to establish a shared infrastructure for multilingual and multimodal question answering in the tourism domain. The QALL-ME system allows users to pose natural language questions in several languages (both in textual and speech modality) using a variety of input devices (e.g. mobile phones), and returns a list of specific answers formatted in the most appropriate modality, ranging from small texts, maps, videos, and pictures. In order to achieve the project's objectives, a domain-specific ontology for the tourism domain was developed and shared among all the partners. The main purpose of the ontology is to provide a common vocabulary for the tourism domain as well as a computerized specification of the meaning of terms used in the vocabulary to enable knowledge sharing and reuse among the partners and succeeding researchers. The ontology will be made incrementally available for research purposes on the project's website, once new versions will be developed. Since the ontology is domain specific, this means that it does not contain general concepts which are applicable to a wide range of domains. In order to address this problem, the QALL-ME ontology was aligned with two upper ontologies, WordNet and SUMO, which are widely used in the NLP field. Due to the changing nature of the QALL-ME ontology throughout the life of the project, automatic means for performing this mapping were investigated. After the ontology was designed, it was used to semantically annotate original data obtained from the tourism web sties and natural language questions created by users. The annotated data was stored in the database as instances of the ontology, and a data access method was provided to retrieve specific content from the database for answering the user's questions.

The subsequent sections are organized as follows: Section 2 reviews the related ontologies developed in the previous projects. Section 3 presents the development of the QALL-ME ontology. Section 4 presents the alignment of the QALL-ME ontology to WordNet and SUMO. Section 5 describes the storage and retrieval of the ontology data, and Section 6 presents the summary and conclusion.

2. Review of Related Ontologies

Recently, a number of research projects focused on the domain of tourism to investigate complex language technologies and web technologies to improve

[1] http://qallme.fbk.eu/

information searching and accessing in this data-rich area. Some important tourism ontologies developed in previous projects are reviewed in this section.

2.1 Harmonise Ontology

The Harmonise ontology was first developed during the Harmonise project and then extended to new sub-domains in the project of Harmonise Trans-European Network for tourism (Harmo-TEN[2]). The two related projects aimed to provide an open mediation service for travel and tourism information exchange within the tourism industry members. The ontology was the centralised data model into which the data models of all participants were mapped. This mapping took place at the side of each individual member, since the mapping between the member's legacy system and the Harmonise ontology was specific to that member. The Harmonise ontology identifies a set of common concepts in the tourism domain for developing a shared, conceptual reference schema in the format of RDFS. It focuses specifically on the following two sub-domains (Clissmann & Höpken, 2004):

- *Events*: conferences, performances, sports, etc.
- *Accommodation*: hotels and guest houses etc., excluding self-catered accommodation such as camping and holiday apartments.

2.2 Hi-Touch Ontology

Hi-Touch is an IST/CRAFT European program, which aimed to develop Semantic Web methodologies and tools for intra-European sustainable tourism. The Hi-Touch ontology was developed mainly by Mondeca[3], using the "Thesaurus on Tourism and Leisure Activities" (World Tourism Organization, 2001) as an authoritative source for its terminology. The ontology focuses on tourism products and customers' tourism expectations. Its usage can ensure the consistency of categorization of tourism resources managed on different distributed databases and enhance searches among numerous tourism products by providing semantic query functionalities. The ontology contains the following three top-level classes (Legrand, 2004):

- A *document* refers to any kind of documentation or advertisement about a tourism product.
- An *object* refers to a tourism offer, which is divided into five subclasses such as *Environment, Activities, Imagination, Ethics,* and *Logistics*.
- A *publication* is a document created from the results of a query – the answers of a query may be combined together to form a PDF document.

2.3 eTourism Ontology

The eTourism Semantic Web portal was developed by Digital Enterprise Research Institute (DERI)[4], Innsbruck, Austria. An ontology was used to provide eTourism vocabulary for annotations and obtain agreement on a common specification language for sharing semantics. It mainly covers accommodation and activities, including also the necessary infrastructure for the activities.

- *Accommodation* classifies all facilities like hotels, guest houses and apartments.
- *Activities* refer to skiing, bowling, snow boarding etc.
- *Infrastructure* refers to those facilities provided for the above activities, such as bowling halls, skiing resorts, riding stables and tennis courts.

The web pages offering information about accommodation and activities were collected from the Web and then converted into machine-readable semantic data in the format of RDF based on the ontology. The eTourism portal developed by DERI consisted of a search interface, where information retrieval was based on the semantic data to allow better requests.

2.4 TAGA Ontology

Travel Agent Game in Agentcities (TAGA)[5] is an agent framework for simulating the global travel market on the Web. In TAGA, all travel service providers can sell their services (e.g. *flights* and *hotels*) on the Web and thus form a Web travel market, and travel agents help customers to buy the travel package from the Web travel market according to the customers' preferences, e.g. taking economy flights, staying near the city centre, or eating Italian food. TAGA defined two domain ontologies to be used in simulations. The first one *(travel.owl*) covers basic travel concepts such as itineraries, customers, travel agents, travel service providers, and service reservations, whereas the second one (*auction.owl*) defines different types of auctions, roles that the participants play in them and the protocols used etc.

2.5 GETESS Ontology

The BMBF funded project – German Text Exploitation and Search System (GETESS) – aimed at developing an intelligent Web tool for information retrieval in the tourism domain. GETESS enabled natural language description of search queries through navigation in a domain-specific ontology and presented the results in an understandable form. The GETESS ontology contains 1043 concepts and 201 relations and provides bilingual terms (English and German) for each concept. It is the central service for text mining, storage and query of semantic content by determining which facts may be extracted from texts, which database schema must be used to store these facts and what information is made available at the semantic level (Staab et al., 1999).

3. Ontology Development

The QALL-ME ontology was developed after a thorough investigation of the above related ontologies; as a result it borrows some concepts and structures from them. In terms of coverage, the QALL-ME ontology is similar to the Harmonise and eTourism ontologies in that they all

[2] http://www.harmo-ten.org/
[3] http://www.mondeca.com/
[4] http://e-tourism.deri.at/
[5] http://taga.sourceforge.net/

focus on static tourism information (e.g. *accommodation* and *events/activities*) rather than dynamic information related to travel business (e.g. *customers* and *itineraries*) as the TAGA and Hi-Touch ontologies do. However, the QALL-ME ontology has a bigger coverage than the two aforementioned ontologies since it includes more types of tourism sites and events. In terms of structure, the QALL-ME ontology is similar to the eTourism ontology, since both of them are written in the Web Ontology Language (OWL) rather than RDFS, thus involving more complex classes and relationships and supporting complex inferences.

For the ontology language, we used OWL, which is the most recent development in standardized ontology languages, endorsed by the World Wide Web Consortium (W3C). Compared with the preceding ontology languages such as RDFS, OIL, DAML+OIL, OWL is equipped with a rich expressive power, has a clear layered structure for scalability and is compatible with its predecessors (Arroyo et al., 2004). OWL has three sublanguages – OWL Lite, OWL DL and OWL Full, for different purposes and with an increasing expressiveness. We selected OWL DL as the encoding language, since it has more expressive power than OWL Lite and more efficient reasoning support than OWL Full (Antoniou & van Harmelen, 2004). In addition, we used Protégé-OWL[6] as the editor and RacerPro[7] as the reasoner.

The QALL-ME ontology aims at providing a conceptualized description of the selected domain. It covers several important aspects in the tourism industry, including *tourism destinations* (i.e. cities and towns), *tourism sites* (i.e. accommodation, gastro, attraction, and infrastructure), *tourism events* (e.g. movie and show) and *transportations*. The ontology contains 122 classes (concepts), 55 datatype properties and 52 object properties which indicate the relationships among the 122 classes. The 122 classes were categorized into 15 top-level classes. The class hierarchy has a maximum depth of 4.

From the point of view of design, the 15 top-level classes fall into the following three categories:

- **Main classes** refer to the most important concepts in the tourism domain;
- **Element classes** refer to the elements of the main classes or the elements of other element classes;
- **Attribute classes** refer to the packages of a group of attributes for the main classes or element classes.

In the ontology, there are six main top-level classes, which are:

- ***Country***
- ***Destination***: tourism destinations, e.g. cities or towns.
- ***Site***: the places providing services for tourists, having four subclasses:
  - *Accommodation*: the places providing accommodation service, having seven subclasses – *Resort*, *Hotel*, *Campsite*, *Hostel*, *Cottage*, *Chalet* and *Bed&Breakfast*.
  - *Gastro*: the places providing prepared food and drinks, having four subclasses – *Restaurant*, *Café*, *Club* and *Bar &Pub*.
  - *Attraction*: the places of interest which tourists visit, having five subclasses – *CulturalHeritage, ReligiousHeritage, NaturalHeritage, ThemePark* and *Zoo&Aquarium.*
  - *Infrastructure*: the places providing infrastructure facilities or services, having several subclasses such as *Cinema*, *Theatre*, *Gym*, *Stadium*, *Pharmacy*, *Hospital*, *PostOffice*, *Bank*.
- ***EventContent***: refers to static information about an event, having nine subclasses – *Movie*, *Show*, *Concert*, *Exhibition*, *Convention*, *Competition*, *Ceremony*, *Activity* and *Party*.
- ***Event***: refers to dynamic information about an event, i.e. occurrence of an event.
- ***Transportation***: the vehicles connecting two terminals, having two subclasses:
  - *IntraCityTransportation:* those inside a destination, having two subclasses – *Bus* and *Metro.*
  - *InterCityTransportation:* those connecting two destinations, having four subclasses – *Flight, Train, Coach* and *Ship.*

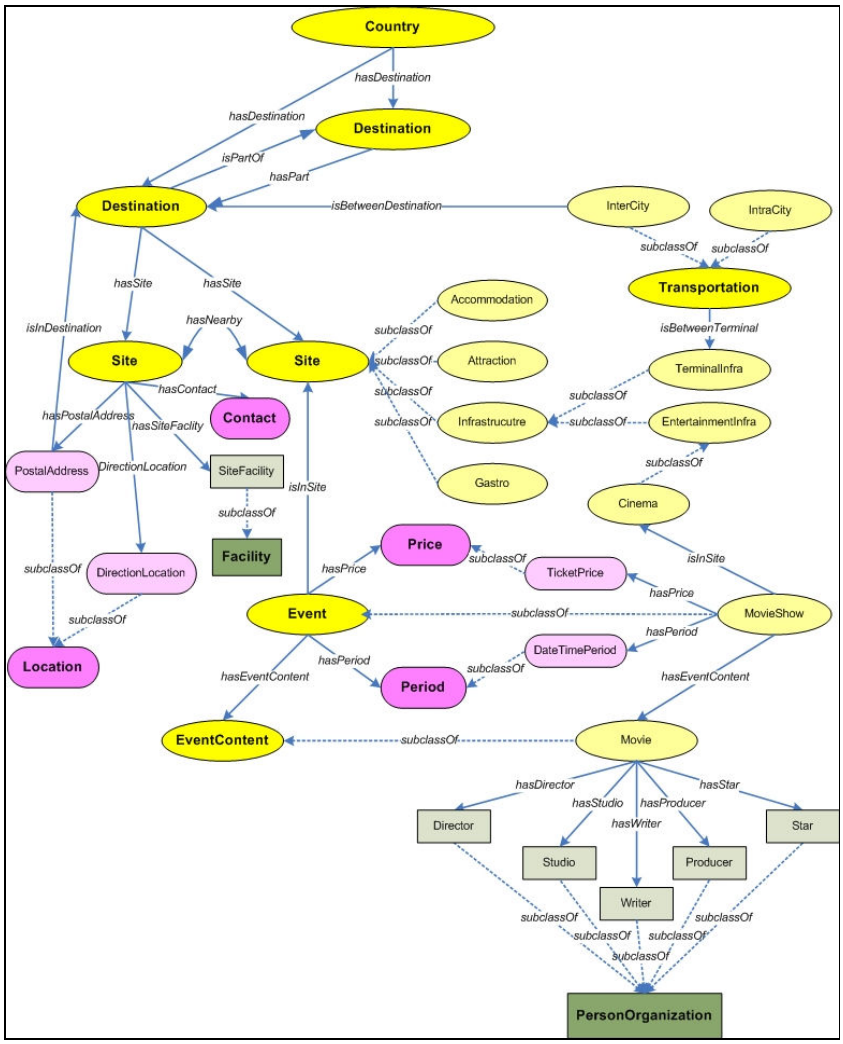
The five element classes defined in the ontology are:

- ***Room***: an inside space in a building, having four classes – *GuestRoom*, *ConferenceRoom*, *RestaurantRoom* and *CinemaRoom.*
- ***Facility***: having two subclasses – *SiteFacility* and *RoomFacility*.
- ***PersonOrganization***: having ten subclasses – *Star*, *Director*, *Performer*, *Writer*, *Competitor*, *Composer*, *Conductor*, *Choreographer*, *Studio* and *ServiceProvider*,
- ***Language***
- ***Currency***

The four attribute classes defined in the ontology are:

- ***Contact***
- ***Location****:* having three subclasses:
  - *DirectionLocation*
  - *PostalAddress*
  - *GPSCoordinate*
- ***Period****:* having three subclasses:
  - *DatePeriod*: from a start date to an end date
  - *TimePeriod*: from a start time to an end time
  - *DateTimePeriod*: a pair of Date period and Time period
- ***Price****:* having three subclasses:
  - *RoomPrice*: having two subclasses – *ConferenceRoomPrice* and *GuestRoomPrice*.
  - *TicketPrice*
  - *GastroPrice*: having three subclasses – *FoodPrice*, *DrinkPrice* and *MealPackagePrice*.

[6] http://protege.stanford.edu/overview/protege-owl.html
[7] http://agraph.franz.com/racer/



- *Ellipse, rectangle, and rounded rectangle boxes respectively represent main classes, element classes, and attribute classes, and highlighted classes in bold are top-level classes.*
- *Solid lines represent properties, and dot lines represent subclasses.*

Figure 1: The main classes and some of their important relationships in the QALL-ME ontology

From the point of application, the instances of attribute classes and element classes cannot exist independently, and must be attached to an instance of the main classes or other element classes. The main classes and their relationships with some element and attribute classes in the QALL-ME ontology are shown in Figure 1.

A country must contain at least one destination (i.e. ∃hasDestination *some* Destination), and a destination must belong to a country (i.e. ∃isInCountry *some* Country). A destination may be a part of a bigger destination (i.e. isPartOf) or may contain several smaller destinations (i.e. hasPart). A destination must contain at least one site (i.e. ∃hasSite *some* Site). A site can be accommodation, gastro, attraction or infrastructure. It may have some nearby sites (i.e. hasNearby). A site may host one or more events (i.e. hasEvent). Transportation is used to connect two terminals which are a kind of infrastructure (i.e. isBetweenTerminal).

A site may contain several elements such as site facilities (i.e. hasSiteFacility) and rooms (i.e. hasRoom). It also has one or more spoken languages on site (i.e. hasSpokenLanguage) and a service provider (i.e. hasServiceProvider) which is either a person or an organization. The properties of a site include contact information (i.e. hasContact) and location information which is divided into direction information (i.e. hasDirectionLocation), postal address (i.e. hasPostalAddress), and GPS coordinate (i.e. hasGPSCoordinate). A site also has opening times (i.e. hasOpeningTime) which are time periods or date periods, or a combination of a pair of date period and time period. In addition, a site may have prices (i.e. hasPrice) for its specific services or products. The price is often related to the date or time period. For example, a hotel has a specific guestroom price for a type of guestroom (i.e. hasRoom) in a specific date period (i.e. hasPeriod).

An event must contain a specific kind of content (i.e. hasEventContent) and happen in a site (i.e. isInSite) at a specific time on a specific date (i.e. hasPeriod), and it sometimes has a price for admission (i.e. hasPrice). For the event of movie show, the site is Cinema, the content is Movie, the price is TicketPrice and the period is DateTimePeriod. A date&time period contains a pair of date period (i.e. hasDatePeriod) and time period (i.e. hasTimePeriod). A date period is composed of startDate and endDate, whereas a time period is composed of startTime and endTime.

4. Ontology Alignment

The QALL-ME ontology was designed as a model of the narrow knowledge domain of tourism. To provide for broad coverage of data over which automatic ontology-based reasoning can be carried out, it was necessary to map it to the upper ontologies. Given the dynamic nature of the ontology and the data that it is expected to cover, we investigated semi-automatic methods to perform ontology alignment.

The task of ontology alignment can be described as follows: given two ontologies, each of which describes a set of discrete elements (*classes*, *properties*, *rules*, etc.), find the relationships (*equivalence* or *subsumption*) holding between these elements. In the QALL-ME project, the upper ontologies chosen to complement the QALL-ME ontology are WordNet 2.1 and SUMO, both of which are widely used for various broad-coverage general-domain NLP applications. Our alignment algorithm relied on NLP and structural similarity techniques. The ontology alignment procedure consists of the following stages:

- **String similarity of element identifiers**

The QALL-ME class identifiers are single words (e.g. *Location*) or multi-word phrases (e.g. *PostalAddress*). The multi-word identifiers were split into individual words, each of which was part-of-speech tagged, and a simple heuristic was used to find the heads of the phrases and convert the heads into their canonical form. The WordNet synsets that correspond to ontological classes have numeric identifiers, so the words inside the synset were used for string matching. Exact matches were taken to indicate *equivalency* between the QALL-ME class and WordNet synset. If only the head of a multi-word QALL-ME identifier was matched to a WordNet synset, the correspondence was taken to be that of *subsumption*, assuming that multi-word expressions denote more specific concepts than their heads.

- **Structural similarity for disambiguation**

Single-word QALL-ME identifiers and the heads of multi-word ones that matched more than one synset in WordNet were disambiguated by measuring their structural similarity to each candidate WordNet synset. The measure of similarity between a QALL-ME concept c_q and a WordNet synset c_w was based on the idea that if c_q and c_w were to match, they should be at equal semantic distances from other concepts already known to unambiguously correspond, C_q^m and C_w^m. The similarity score between c_q and c_w is computed by first determining the normalized distances[8] between c_q and each concept in C_q^m. The same distances are computed for c_w. The ratio between the distances to already matching concepts is then calculated, and the final similarity score is obtained as the average of these ratios.

- **Definition similarity for disambiguation**

If certain QALL-ME concepts are still not disambiguated because neither their parents nor descendants have been previously matched, the algorithm aims to decide among the candidate WordNet synsets by measuring the Jaccard similarity (Manning & Schuetze, 1999) between the definitions of the concepts. Our algorithm gives a lower priority to the definition similarity than to the structural similarity because whereas WordNet synsets are supplied with dictionary-like glosses, QALL-ME concepts do not have definitions as such, but rather clarifying comments; also, only some of the concepts have such comments. The reason for this is that writing concept definitions is a time-consuming process and rather impractical, given the dynamic nature of the ontology and the fact that relevant definitions can be obtained from WordNet via ontology alignment.

- **Structural similarity for unmatched concepts**.

At this point a few QALL-ME concepts still remain unmatched – either those whose head of the identifier does not correspond to any word in WordNet (e.g. *TransportInfra*) or those that could not be disambiguated because none of their parents and descendants were previously aligned. To find the most likely equivalent for these concepts, we measure their structural similarity to all WordNet noun synsets.

From the calculated alignment between the QALL-ME ontology and a given version of WordNet, we obtained QALL-ME alignments to SUMO and different versions of WordNet (from 1.6 and up) using publicly available mapping files. Since WordNet had been mapped with EuroWordNet[9], a multilingual resource, the QALL-ME ontology was therefore linked to the multilingual terms.

	Correct	Incorrect	Failed to match
Single WN match	23	0	0
Multiple WN matches	10	34	32
No WN matches	0	0	9
Total	33	34	41

Table 1: The accuracy of the alignment

[8] The semantic distance between two concepts within an ontology was measured using Learning Accuracy (Hahn and Schattinger 1998).

[9] http://www.illc.uva.nl/EuroWordNet/

QALL-ME	SUMO	WN2.1	WN2.1 Gloss
Accommodation	@inhabits	=02647858	living quarters provided for public convenience; "overnight accommodations are available"
Chalet	@Building	=02973228	a Swiss house with a sloping roof and wide eaves or a house built in this style
GasStation	@Corporation	=03388513	a service station that sells gasoline
Location	=located	=00026074	a point or extent in space
PostOffice	@Organization	=08034771	an independent agency of the federal government responsible for mail delivery
Ship	@TransportationDevice	= 04145707	a vessel that carries passengers or freight

Table 2: Sample alignments between QALL-ME, WordNet 2.1 and SUMO

To evaluate the alignment quality of this algorithm, we compared the automatically generated alignments against the manually produced alignments between the two ontologies, which were considered as a gold standard. Table 1 summarises the results of the evaluation.

While the overall accuracy of the alignment is clearly suboptimal (precision of 49% and recall of 31%), we noticed that for concept names with unambiguous matches in WordNet the algorithm performs without any errors. Ambiguous concept names are the major source of errors. Concept names without any WordNet matches are relatively few, but they also cannot be reliably assigned to a WordNet synset. An evident reason for poor disambiguation quality seems to be the fact that at the moment the QALL-ME ontology is rather shallow (maximum depth is 4 edges), which, even after distance normalisation, does not provide reliable estimates of the semantic distance between concepts. Furthermore, only a few concepts have comments that are usable for definition similarity.

Based on that, we developed a semi-automatic alignment procedure, whereby unambiguous matches were aligned automatically, multiple WordNet matches were disambiguated by a human, and concept names without any matches were aligned completely manually. Table 2 gives some examples of the alignments we obtained ('=' indicates *equivalence* and '@' *subsumption*). We expect that if the QALL-ME ontology acquires a deeper structure and more comments are added to its concepts, the disambiguation step might yield better accuracy and eventually the disambiguation step can be partially automated.

5. Storage and Retrieval of Ontology Data

To make an ontology applicable in a real system, an underlying data model is needed to store records about the things defined in the ontology. There are different ways of storing and encoding sets of records. We used a graph model, RDF, which represents data as a collection of triples in the form of <subject, predicate, object>. Once the QALL-ME ontology was developed and aligned, the original tourism data, which are possible answers to user questions, were annotated using the ontology and represented as the RDF triples. For the structured data obtained in XML or database formats, a mapping between XML tags or table fields to the classes and relations in the ontology was created and used to convert the structured data in other formats into the RDF format directly. For the unstructured data which were obtained as HTML pages taken from tourism web sites, a set of wrappers were developed to locate snippets of useful information, and annotate them semantically using XML tags. For each kind of data (e.g. *Hotel, Cinema* and *Movie*) which corresponds to a class in the ontology, an XML schema was designed according to the OWL ontology. The XML files for storing this kind of data must conform to the corresponding XML schema. Finally, the data was transformed from the XML format to the RDF format using XSLT, and thus the RDF data became the instances of the QALL-ME ontology. The annotation procedure is shown in Figure 2.

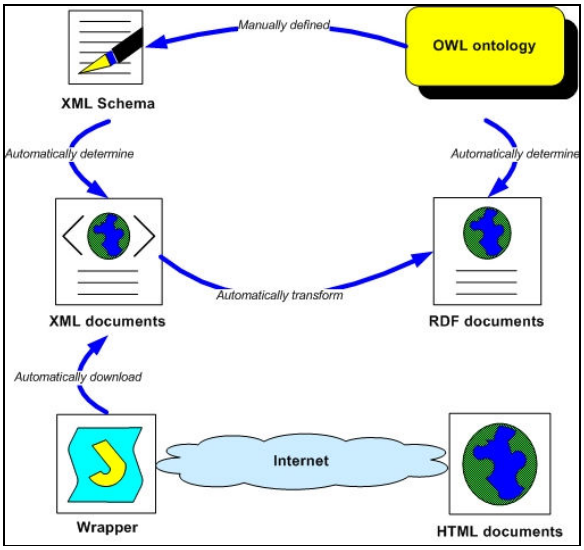


Figure 2: The procedure for data annotation

For storing the RDF data, various storage systems have been implemented with different methods and targets and support different query languages. These include Jena2 (Wilkinson et al., 2003), Sesame (Broekstra, Kampman & van Harmelen, 2002), Kowari (Wood, Gearon & Adams, 2005), KAON (Bozsak et al., 2002).

For the application of question answering, response time is a crucial issue, especially for the enormous amount of data. Thus the storage system must have good scalability and efficiency. Furthermore, the system should provide sufficient reasoning capability to support the semantic requirements of the application. Considering the above requirements, we selected Jena2 as the storage system and the SPARQL query language as the data retrieval method. Jena2 has some unique features (e.g. denormalized database schema, arbitrary property tables, optimized storage for reification), which make it an efficient storage solution (Wilkinson et al., 2003). SPARQL is a RDF query language, which has become a WC3 recommendation (Prud'hommeaus & Seaborne, 2008).

Jena2 supports in-memory storage and database storage. Storing data in a file system is the simplest but also the most inefficient technique for storing data because the whole file has to be scanned and all the data has to be loaded into the computer's main memory for each-time running. It is quite efficient for a small amount of data but not able to handle large data. We therefore imported the RDF data into a MySQL-persistent database using Jena2. In contrast to the in-memory storage, the database-persistent storage saves the overhead of loading the data into the model each time, and the RDF models significantly larger than the computer's memory can be stored too. The latter, however, comes at the expense of a higher overhead (a database interaction) to retrieve and update RDF data from the model. Figure 3 shows the architecture for data storage and data access using Jena2.

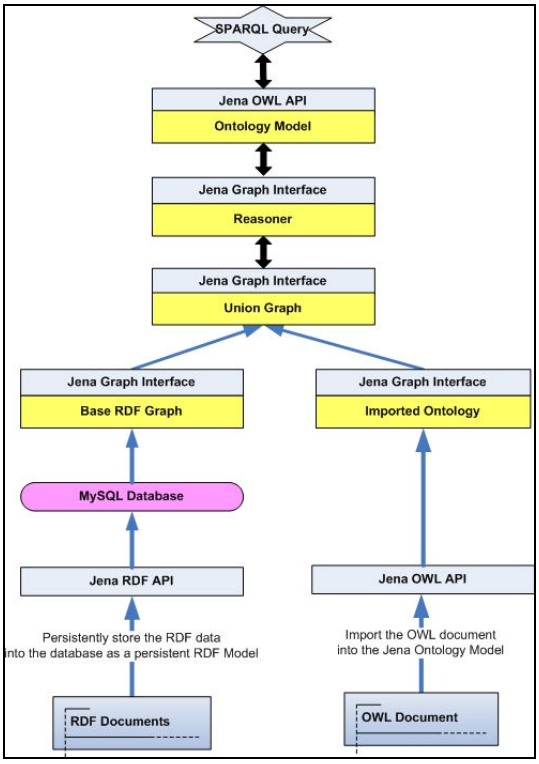


Figure 3: Data storage and data access using Jena2

The RDF documents were loaded into the RDF model, which was then persistently imported into the MySQL relational database, through the Jena RDF API. The RDF data stored in the database is called *Base RDF Graph* which holds the asserted statements. The OWL ontology document which only defines the classes and properties in the domain is imported from the file system through the Jena OWL API. The OWL model, called *Imported Ontology*, was not persisted in the database because the meta-model will often be updated and the file size is very small and easy to load. The *Base RDF Graph* and the *Imported Ontology* form the *Union Graph*. Jena's *Reasoner* can use the contents of the *Base Graph* and the semantic rules of the OWL language to derive additional statements that the model does not express explicitly. SPARQL queries were operated on the Ontology *Model* to retrieve specific pieces of data from it.

To extract answers from the RDF database, a natural language question needs to be transformed into a SPARQL query. It is a difficult task involving complex semantic annotation of the question with the use of the defined ontology. Various methods to automatically perform such annotation and transformation are being investigated in the QALL-ME project, e.g. Negri, Magnini & Kouylekov (2008) and Ou et al. (2008), which are out of the scope of this paper. A simple example is given in Table 3 to show how to annotate the words and phrases in the question using the concepts and properties defined in the ontology and create a SPARQL query.

6. Summary and Conclusion

The paper presents the development of a tourism ontology which aims to support multilingual Question Answering. In contrast to other ontologies in the same domain, this ontology has a bigger scope in that it covers more aspects of the domain. The ontology was encoded in OWL DL, an ontology language having richer expressive and reasoning power than other ontology languages. Furthermore, the ontology was aligned with WordNet and SUMO, two upper ontologies, to expand concepts defined in the domain.

The tourism data was mainly obtained as HTML web pages, and semantically annotated based on the ontology and thus converted into the RDF format as the instances of the ontology. We stored the annotated data persistently in a MySQL database using Jena2. The SPARQL query language was used to access and manipulate the annotated data. In the future, we will explore other storage solutions (e.g. Jena2's SDB [10] and Sesame) and compare their efficiency. Natural language questions can be annotated using the aligned ontology and transformed into SPARQL queries to retrieve answers from the database. Various methods performing such annotation and transformation are being investigated in the QALL-ME project.

[10] http://jena.hpl.hp.com/wiki/SDB

Question	*What is the **film** name which has **star** Halle Berry and is being **shown** in Birmingham?*
Annotation	Movie <Movie: name> Star <Star: name> MovieShow <Destination: name>
SPARQL Query	**SELECT** ?movieName **WHERE {** ?MovieShow rdf:type prefix:MovieShow. ?MovieShow prefix:isInSite ?Cinema. ?Cinema prefix:hasPostalAddress ?PostalAddress. ?PostalAddress prefix:isInDestination ?Destination. ?Destination prefix:name "*Birmingham*"^^<xsd:string>. ?MovieShow prefix:hasEventContent ?Movie. ?Movie prefix:name ?movieName. ?Movie prefix:hasStar ?Star. ?Star prefix:name "*Halle Berry*"^^<xsd:string>. **}**

Table 3: Semantic annotation of a question based on the aligned QALL-ME ontology and its SPARQL query

7. Acknowledgements

This work is supported by the EU-funded project QALL-ME (FP6 IST-033860).

8. References

Antoniou, G. & van Harmelen, F. (2004). Web Ontology Language: OWL. In S. Stabb & R. Studer (Eds.), Handbook on Ontologies. Heidelberg, Germany: Springer, pp. 67-92.

Arroyo, S., Lara, R., Ding, Y., Stollberg, M. & Fensel, D. (2004). Semantic Web Languages: Strengths and Weakness. In Proceedings of the International Conference in Applied Computing.

Bozsak et al. (2002). KAON: Towards a Large Scale Semantic Web. In Proceedings of the 3rd International Conference on Electronic Commerce and Web Technologies. Heidelberg, Germany: Springer, pp. 304-313.

Broekstra, J., Kampman, A. & van Harmelen, F. (2002). Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema. In Proceedings of the 2nd International Semantic Web Conference. Heidelberg, Germany: Springer, pp. 54-68.

Clissmann, C. & Höpken,W. (2004). Harmonise Ontology User Manual, Retrieved 5 March, 2008, from http://www.harmo-ten.info/harmoten_docs/D2_2_Ontology_User_Manual_V3.2.0.3.doc

Fellbaum, C. (ed.) (1998). WordNet, An Electronic Lexical Database. Cambridge, MA: The MIT Press.

Gruber. T. R. (1993). A translation approach to portable ontologies. Knowledge Acquisition, 5(2), pp. 199-220.

Hahn, U. & Schattinger, K. (1998). Towards Text Knowledge Engineering. In Proceedings of the 15th National Conference on Artificial Intelligence. Menlo Park, CA: AAAI Press, pp. 524-531.

Legrand, B. (2004). Semantic Web Methodologies and Tools for Intra-european Sustainable Tourism. White paper. Paris: Mondeca.

Lenat, D. B. (1995). Cyc: A large-scale investment in knowledge infrastructure. Communication of the ACM, 38(11), pp. 33-38.

Manning, C. & Schuetze, H. (1999). Foundations of Statistical Natural Language Processing. Cambridge, MA: The MIT Press, pp. 299.

Negri, M., Magnini, B. & Kouylekov, M. (2008).Detecting Expected Answer Relations through Textual Entailment. In Proceedings of the 9th International Conference on Intelligent Text Processing and Computational Linguistics. Heidelberg, Germany: Springer, pp. 532-543.

Niles, I. & Pease, A. (2001). Towards a Standard Upper Ontology. In Proceedings of the 2nd International Conference on Formal Ontology in Information Systems. New York, NY: ACM, pp. 2-9.

Ou, S., Orasan, C., Mekhaldi, D. & Haslser L. (2008). Automatic Question Pattern Generation for Ontology-based Question Answering. In Proceedings of the 21st International Florida Artificial Intelligence Research Society Conference. Menlo Park, CA: AAAI Press.

Prud'hommeaux, E. & Seaborne, A. (2008). SPARQL Query Language for RDF. WC3 Recommendation 15 January 2008. Retrieved 14, March, 2008, from http://www.w3.org/TR/rdf-sparql-query/.

Staab, S. et al. (1999). GETESS – Searching the Web Exploiting German Texts. In Proceedings of the 3rd International Workshop on Cooperative Information Agents III. Heidelberg, Germany: Springer, pp. 113-124.

Wilkinson, K., Sayers, C., Kumo, H. & Reynolds, D. (2003). Efficient RDF Storage and Retrieval in Jena2. In Proceedings of the 1st International Workshop on Semantic Web and Databases. Berlin, Germany: Humboldt-Universität, pp. 131-150.

Wood, D., Gearon, P. & Adams, T. (2005). Kowari: A Platform for Semantic Web Storage and Analysis. In Proceedings of XTech 2005 Conference.

World Tourism Organization. (2001). Thesaurus on Tourism & Leisure Activities. Madrid, Spain: World Tourism Organization.