Quelo @ The IESD Challenge 2012

Enrico Franconi, Paolo Guagliardo, and Sergio Tessaris

Free University of Bozen-Bolzano

Abstract. *Quelo* is a tool aimed at the exploration and querying of semantic data by means of an intelligent visual interface based on natural language. The tool is not tied to any specific application domain or class of users, and no advance knowledge of the ontology and its vocabulary is indeed required in order to use it, which makes Quelo a general-purpose tool suitable for a wide range of users, whether they are domain experts (e.g., ontology engineers) or not. Knowledge of the ontology is gradually acquired by using the tool itself, through an iterative process of information discovery. Users are guided step by step in constructing a query over the ontology that best captures their information need, and they are supported in this task by Quelo's intelligent interface that provides constant contextual feedback and visual assistance. By using automated reasoning techniques, Quelo filters out contradictory and/or redundant pieces of information w.r.t. the ontology, lifting users from the worry of making "wrong" and/or ineffective choices when navigating through the semantic structures or building their queries.

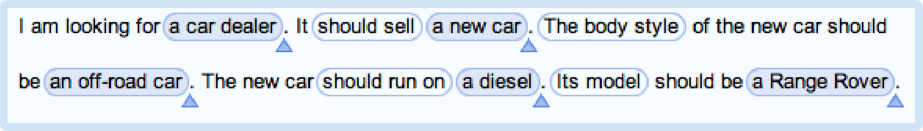
1 Introduction and Overview

Quelo is a visual tool that supports users in formulating a precise query (or statement) over an ontology by means of an intuitive interface based on natural language. Queries are represented as a coherent sequence of snippets of natural language text – i.e., forming a meaningful English sentence – and built through a refinement process, called *intensional navigation* of the ontology, entirely guided by the system. This process consists in the iteration of few basic operations, starting from a generic request, until the result satisfies the user's information need. By exploiting automated reasoning over the ontology, Quelo's intelligent interface proposes only refinements of the query that are relevant and meaningful in the current context (i.e., the query built so far) w.r.t. the ontology. This contextual feedback is presented to the user by means of snippets of natural language text which can be incorporated into the query; this makes user's choices in proceeding with the intensional navigation clearer and simpler and it is the most powerful and innovative of Quelo's features. Another crucial aspect is that users are not required to possess any specific knowledge of the ontology and its vocabulary; indeed, this can be gradually acquired by using the tool for exploring the ontology, rather than actually querying it, with the aim of discovering general information about the modelled domain. The ideas behind Quelo and its theoretical foundations have been previously investigated in [1, 4, 2, 3, 9, 8, 6, 5, 7]. A demo of Quelo is freely accessible online at the following address:

`http://krdbapp.inf.unibz.it:8080/quelo/`

2 Core Functionality and User Interface

In Quelo's user interface (UI), a query is represented as a sequence of snippets of natural language text composing an English sentence. An example of query as displayed by Quelo is shown here:



I am looking for a car dealer. It should sell a new car. The body style of the new car should be an off-road car. The new car should run on a diesel. Its model should be a Range Rover.

Some of the snippets, corresponding to classes and properties from the ontology, are active and allow to operate on the query. In order to distinguish them, active snippets are enclosed in boxes with rounded corners, having a light-blue background if the snippet corresponds to a class and a transparent one if it corresponds to a property. When *hovering* on active snippets, Quelo provides visual hints about the structure of the query, by highlighting snippets related with the one on which the hovering occurs.

The *selection* is a special region within the query, which is composed of snippets corresponding to classes and is initially empty. Snippets in the selection are displayed inside boxes with a dark-blue background and white text. Snippets are added to or removed from the selection by clicking on them, which toggles selection on and off. The selection can be cleared by clicking on an empty point within the query area.

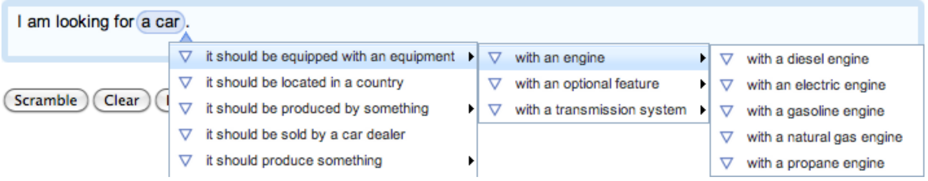
The intensional navigation of any ontology in Quelo starts from the generic query "I'm looking for something". This initial query is then iteratively manipulated through the application of three main operations: *insertion*, for adding new classes and properties; *deletion*, for removing the content of the selection from the query (achieved by dragging the selection outside of the query area); and *substitution*, for replacing the content of the selection with a more general, equivalent or more specific class.

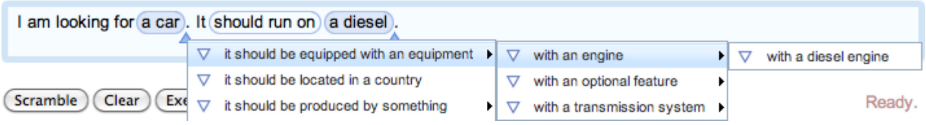
Clicking on one of the triangles appearing below the query baseline brings up a menu with all the classes and properties that can be inserted in that point in the current context. The insertion menu is organised in sub-menus according to the taxonomic information defined in the ontology and it can be easily navigated in order to choose the desired level of detail for the insertion.

Right-clicking on the selection brings up the menu showing all the classes with which the selection can be substituted in the current context. The substitution menu is also organised in sub-menus according to the taxonomic information in the ontology, but it only contains classes and is partitioned in three parts: in the middle are classes that are equivalent to the selection, at the top classes that are more general and at the bottom classes that are more specific.

Each of the active snippets corresponding to properties can be marked as *sticky* by clicking on it. Instances of the classes appearing between a sticky snippet and the next insertion point (i.e., triangle) are included in the answer to the query when this is executed.

Detailed examples illustrating how each of the operations for query refinement works, how to make selections, and other tips and shortcuts, can be found in Quelo's online help pages. What we want to stress here is how the behaviour of Quelo's interface, by dynamically adapting to different contexts, can provide useful insights into the explored ontology. Consider the following example:


I am looking for a car.
it should be equipped with an equipment
it should be located in a country
it should be produced by something
it should be sold by a car dealer
it should produce something
with an engine
with an optional feature
with a transmission system
with a diesel engine
with an electric engine
with a gasoline engine
with a natural gas engine
with a propane engine
Scramble
Clear

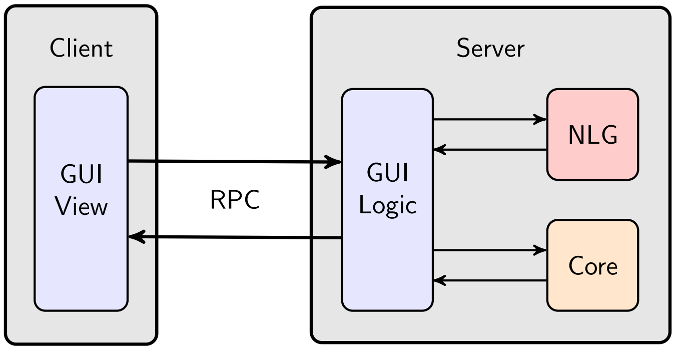

I am looking for a car. It should run on a diesel.
it should be equipped with an equipment
it should be located in a country
it should be produced by something
with an engine
with an optional feature
with a transmission system
with a diesel engine
Scramble
Clear
Exe
Ready.

From the change of content in the insertion menu shown above, users learn that, w.r.t. the ontology under consideration, cars can have engines of several different types, but cars running on diesel can only be equipped with a diesel engine.

The *natural language interface* of the tool hides the composition of a precise query behind the composition of an English sentence describing an equivalent information need. For this interface to work with a specific ontology, a lexicon and a template map must be provided. Devising such resources requires an understanding of the domain along with linguistic notions such as verb tenses, noun genders and countability. To ease the burden of developing these resources, we let the system generate automatically all the necessary resources needed to configure the natural language interface to be used with a new ontology, by using as a source of data the ontology itself, in a completely *domain-independent* fashion.

3 System Design and Architecture

Quelo is web application based on the following client-server architecture and consisting of the three components discussed below.



1. The query logic (a.k.a. *core*) is where the main intelligence of Quelo resides. The core is responsible for all the reasoning tasks that are carried out over the ontology in order to determine which information is relevant w.r.t. the current query [8, 7].
2. The natural language generation (NLG) engine, given a lexicalisation map for the ontology, renders the query into a well-formed English sentence; the lexicon is automatically generated from the ontology [9, 6].
3. The graphical user interface (GUI) provides visual access to the query and to the editing facilities for it, allowing the interaction with the intelligence in the core while benefiting of the services of the NLG engine [5]. The communication between the GUI Logic, residing on the server, and the GUI View, residing on the client (i.e., the user's browser) happens through the mechanism of Remote Procedure Calls (RPC).

At present, Quelo supports ontologies in any language that could be handled by an OWL engine [10], but its extensible design allows for conceptual schemas in other formats. The ontology is fed into the system by just referring to its legal URI. Quelo is as scalable as it is the OWL engine used in the background, being all the services provided polynomially reducible to standard OWL reasoning [8].

References

1. Catarci, T., Dongilli, P., Di Mascio, T., Franconi, E., Santucci, G., Tessaris, S.: An ontology based visual tool for query formulation support. In: Proceeedings of the 16th European Conference on Artificial Intelligence (2004)
2. Catarci, T., Dongilli, P., Di Mascio, T., Franconi, E., Santucci, G., Tessaris, S.: Usability evaluation tests in the SeWAsIE (SEmantic Webs and AgentS in Integrated Economies) project. In: Proceedings of the 11th International Conference on Human-Computer Interaction (2005)
3. Dongilli, P., Franconi, E.: An intelligent query interface with natural language support. In: Proceedings of the 19th International Florida Artificial Intelligence Research Society Conference (2006)
4. Dongilli, P., Franconi, E., Tessaris, S.: Semantics driven support for query formulation. In: Proc. of the 2004 Int. Workshop on Description Logics (2004)
5. Franconi, E., Guagliardo, P., Trevisan, M.: An intelligent query interface based on ontology navigation. In: Proceedings of the Workshop on Visual Interfaces to the Social and Semantic Web (2010)
6. Franconi, E., Guagliardo, P., Trevisan, M.: Quelo: a NL-based intelligent query interface. In: Pre-Proceedings of the Second Workshop on Controlled Natural Languages. vol. 622. CEUR-WS.org (2010)
7. Franconi, E., Guagliardo, P., Trevisan, M., Tessaris, S.: Quelo: an ontology-driven query interface. In: Proc. of the 2011 Int'l Workshop on Description Logics (2011)
8. Guagliardo, P.: Theoretical foundations of an ontology-based visual tool for query formulation support. Tech. Rep. KRDB09-5, Free University of Bozen-Bolzano (Oct 2009), http://www.inf.unibz.it/krdb/pub/TR/KRDB09-05.pdf
9. Trevisan, M.: A portable menu-guided natural language interface to knowledge bases. Master's thesis, University of Gröningen (2009)
10. W3C: OWL 2 Web Ontology Language: Document Overview (Oct 2009), http://www.w3.org/TR/owl2-overview/