Translating Wh-questions into F-Logic Queries

Philipp Cimiano

Institute AIFB
University of Karlsruhe
cimiano@aifb.uni-karlsruhe.de

Abstract

For many applications it is desirable to query knowledge stored in a database or knowledge base through a natural language interface. In this paper we present a Montague-style compositional semantics approach in order to map natural language wh-questions into F(rame)-logic Queries. The semantic representation is constructed on the basis of Lexicalized Tree Adjoining Grammar (LTAG)-style derivation trees.

1 Introduction

Within computational linguistics, the access of knowledge stored in a knowledge base or database through a natural language interface is an important research problem which has received special attention in the mid 80's (see [Androutsopoulos et al.(1995)] for a good introduction and overview of related work). Certainly it is possible to query a knowledge base by using some logical query language, but it is not feasible to assume that non-computer-scientists will find such a language intuitive to use. Another option is to make use of boolean queries such as those used in WWW query interfaces as for example Google or Altavista. This sort of queries are certainly much more intuitive than logical ones, but suffer from a very reduced expressiveness. In this sense the challenge is to use an expressive logical query language in the background while at the same time hiding the complexity of such a language to the user by allowing him to formulate queries in natural language.
In this paper we present a Montague-style compositional semantics approach in order to translate natural language wh-questions into logical form. In particular, motivated by the growing importance of object-oriented database systems, we present a translation into F(rame)-logic [Kifer et al.(1995)]. F-Logic is a fully-fledged first order logic with a model-theoretic semantics. The logic was originally defined to account for the logical properties of object-oriented

systems such as frames, inheritance etc.
The compositional approach to semantic construction is based on *Lexicalized Tree Adjoining Grammar* (LTAG) [Joshi and Schabes(1997)]. LTAG is interesting in this context due to its 'extended domain of locality', i.e. its natural treatment of subcategorization. In particular, in this paper we consider the declarative and axiomatized version of LTAG in [Muskens(2001)] which was extended to incorporate ontological knowledge in [Cimiano and Reyle(2003)]. The semantic representation is constructed on the basis of derivation trees as in [Gardent and Kallmeyer(2003)] with the main difference that we will not use a flat semantic representation nor make use of unification to substitute functional application.
The remainder of this paper is organized as follows: in Section 2 we give a brief overview of F-Logic. In Section 3 we then present the semantic construction approach and illustrate it with some examples. Section 4 presents a conclusion and describes some further work.

2 F-Logic in a Nutshell

F(rame)-logic is a formalism that accounts in a clean and declarative fashion for most of the structural aspects of object-oriented and frame-based languages. F-logic is a fully-fledged first-order logic; it has a model-theoretic semantics and a sound and complete proof theory. In spite of some higher-order syntax constructs, the underlying semantics formally remains first-order in the sense that variables do not range over complex domains.
F-logic incorporates a built-in notion of inheritance. In particular, ":" is used to represent class membership and "::" to denote the subclass relationship. The fact that *Microsoft* is a company would for example be represented as *microsoft:company* and the fact that a company is a thing as *company::thing*. It is interesting to notice that classes are reified, i.e. they belong to the same domain as individual objects. Thus, we can manipulate classes and objects in the same language.
Furthermore, we can specify that a class has certain attributes by using the following syntax: *class[attribute* $\Rightarrow$ *class]*. In addition, these attributes can be parametrized, i.e. we can write *class[attribute@class* $\Rightarrow$ *class]*. Thus, we would model the fact that companies have a name, a boss, a location as well as different revenues every year as follows:

$$\begin{aligned} \text{company}[&\text{name} \Rightarrow \text{string};\\ &\text{boss} \Rightarrow \text{person};\\ &\text{revenue@year} \Rightarrow \text{integer}] \end{aligned}$$

It is important to mention that in F-logic attributes are partial functions, i.e. a certain attribute may not be defined for all the members of the class.

Now in order to define an object of type company, we will basically replace the $\Rightarrow$ by a $\rightarrow$ and replace the class names by concrete values, i.e.

microsoft:company[name $\rightarrow$ "Microsoft";
boss $\rightarrow$ bill_gates;
revenue(2002) $\rightarrow$ 28.370.000.000]

Furthermore, we could now define the *boss* class as well as a concrete instance of it:

boss[name $\Rightarrow$ string;
own $\Rightarrow$ company]

bill_gates:boss[name $\rightarrow$ "Bill Gates";
own $\rightarrow$ microsoft]

In addition, we can also define rules. The following ones state for example that if somebody owns a company, he is also the boss of that company and the other way round: $B[owns \rightarrow C] \leftrightarrow C[boss \rightarrow B]$.
Finally and most important for the work presented in this paper, we can specify queries. The following query can for example be paraphrased as *Which boss owns Microsoft?*:

$$? - X : boss \wedge X[own \rightarrow microsoft]$$

The following query asks for Microsoft's revenue in 2002:

$$? - microsoft[revenue(2002) \rightarrow X]$$

For further details the interested reader is referred to [Kifer et al.(1995)].

3 Semantic Construction

We describe our approach to semantic construction on the basis of the following examples:

(1) Who owns Microsoft?

(2) Which was the revenue of Microsoft in 2002?

(3) Which company does Bill Gates own?

As already mentioned in the introduction, in order to translate wh-questions into F-logic queries we make use of a Montague-style compositional semantics approach. As underlying syntactic theory we build on *Lexicalized Tree Adjoining Grammar* (LTAG) [Joshi and Schabes(1997)]. LTAG is especially interest-

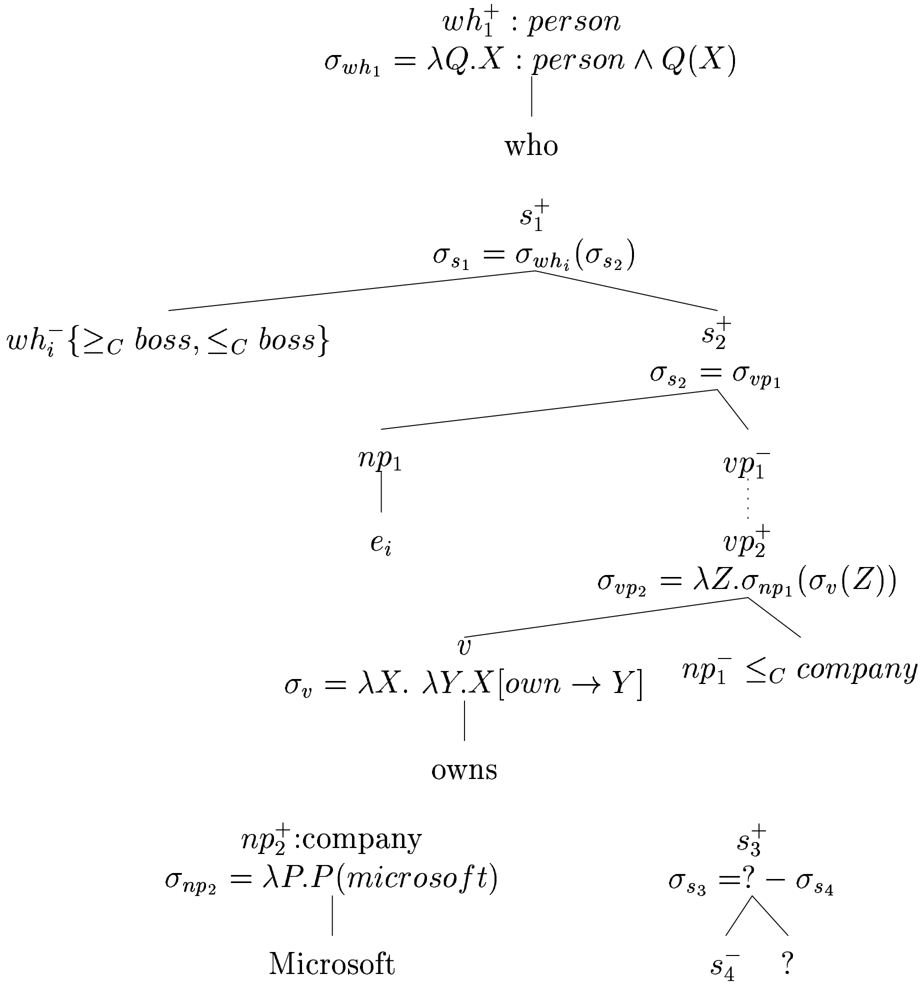

Fig. 1. Elementary trees for *Who owns Microsoft?*

ing in this context because of its extended domain of locality and thus the natural way in which subcategorization is treated. However, we extend the LTAG formalism to also include ontological information. In [Cimiano and Reyle(2003)] we show in fact how the declarative and axiomatized LTAG version in [Muskens(2001)] can be extended by integrating an ontological hierarchy as well as allowing substitution only if the ontological categories of the nodes are compatible. Furthermore, in our LTAG version it is possible to specify for a certain node the head of the node to be substituted. The elementary trees of example 1 are given in Figure 1. Our syntactic analysis of wh-questions is in line with the one presented in [Gardent and Kallmeyer(2003)] in the sense that for each verb there will be one elementary tree for each argument that we can ask for. In line with Muskens' approach, we distinguish between positively anchored nodes (marked with a $+$) and negatively anchored nodes (marked with a $-$). Positively marked nodes dominate some lexical element, while negatively marked ones do not and thus have to be identified with a positively marked one. Nodes which are positively and negatively anchored are said to be *saturated* and have no mark. As in [Muskens(2001)], parsing of a sentence thus boils down to identifying positively and negatively marked nodes in a one-to-one fashion. The $\leq_C/\geq_C$ marks mean that the node attached to the node in question has to be ontologically equivalent or more special/general. Actually, $\leq_C$ defines a partial order on the set of concepts C relevant for the domain in question (compare [Cimiano and Reyle(2003)]). The ":" sign denotes that the corresponding node belongs to the concept following it.[1] Curly brackets denote alternatives. It is important to mention that a wh-pronoun belongs to the most general ontological category it can represent, i.e. *person* in the case of *who*, *thing* in the case of which or what and *location* in the case of *where*. Accordingly, the elementary tree for *owns* in Figure 1 allows the topicalized wh-element to be ontologically equivalent, more general or more special than the selectional restrictions imposed by the verb on the corresponding position. Thus, assuming that *person* and *thing* are disjoint concepts, *Who/Which person/Which boss owns Microsoft?* are valid questions, while *What/Where owns Microsoft* are not.

The Montague-style semantics ([Montague(1974)]) of each elementary tree is given under the root node. Interestingly, the question mark gets also assigned a semantics, i.e. it makes an F-logic query out of an F-logic formula. If we now identify positive and negative nodes in a one-to-one fashion as in Muskens' approach, we yield the unique syntactic tree for example 1 in Figure 2. The reader may easily verify that the resulting semantics is the one found under the root s node in Figure 2. The reader may also verify that with the standard semantics for determiners such as *a*, *every* etc. we also assign the correct semantics to questions such as *Who owns a company?* or *Who owns every company?*, i.e. $? - \exists Y\ Y : company \wedge X[own \rightarrow Y]$ and $? - \forall Y\ Y : company \wedge X[own \rightarrow Y]$ respectively.

The elementary trees for example 2 are given in Figure 3. The interesting move here is that *revenue* is modeled as a relational noun which subcategorizes a PP with the preposition *of* and an NP of type *company* as well as a PP with the preposition *in* and an NP of type *year*. So, here we have made use of the fact that we can specify the head of the constituent which attaches 'below' a given constituent. This is certainly important when dealing with a more frame-based representation such as F-Logic. It is furthermore important to note that the fact that *the* introduces a definite description is ignored here. It is easy to verify that mapping nodes to each other in a 1-1 fashion yields the correct syntactic tree as well as that the semantics is as expected, i.e. $\sigma_s =? -\ X : thing \wedge microsoft[revenue(2002) \rightarrow X]$.

Example 3 is in principle dealt with as example 1 except for the fact that we need an alternative elementary tree for *which* taking an NP as complement

[1] The ':' sign obviously alludes to the same sign used in F-logic but is used here to denote the assignment of nodes to ontological concepts

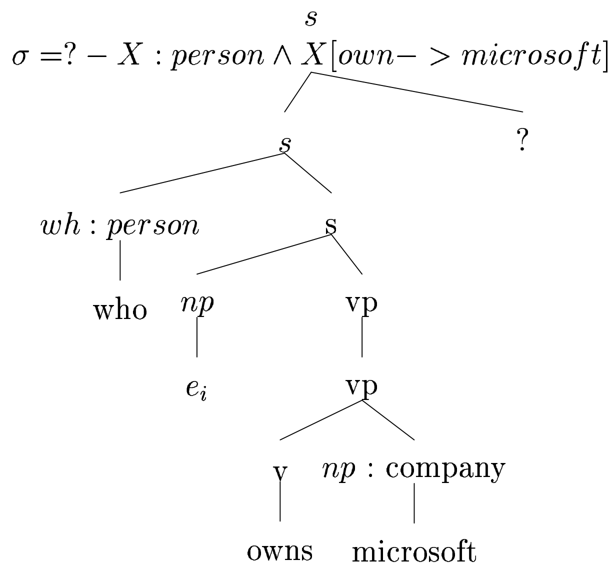

Fig. 2. Resolution of *Who owns Microsoft?*

(see figure 4). Furthermore, we need an additional elementary tree for *own* in which the object position (the company) is topicalized. The elementary tree for the auxiliary *does* is adjoined as usual. The final semantics then looks as follows: $\sigma_s =? -\ X : company \wedge bill\_gates[own \rightarrow X]$.
Before concluding this section, we mention one further interesting possibility which our LTAG-based approach allows. In fact, we can start a clarification dialog if some constituent of the question is missing. Imagine for example a user asking *What was the revenue of Microsoft?* Then it will not be possible to map nodes to each other in a 1-1 fashion and the system could ask the user to provide the missing constituent by asking for a constituent with the category of the (still) negatively anchored node as follows: *In which year?* or *Please specify a year.*

4 Conclusion and Further Work

We have presented a compositional semantics approach to translate wh-questions into F-Logic queries. For this purpose, we have assumed the declarative formulation of LTAG in [Muskens(2001)] as underlying syntactic theory. Furthermore, in line with [Cimiano and Reyle(2003)] we have extended the formalism by including information about the ontological class of nodes thus yielding constraints on their possible identification. Furthermore, it has turned out that the use of LTAG-style derivation trees allows to ask back the user for missing information in a straightforward manner.

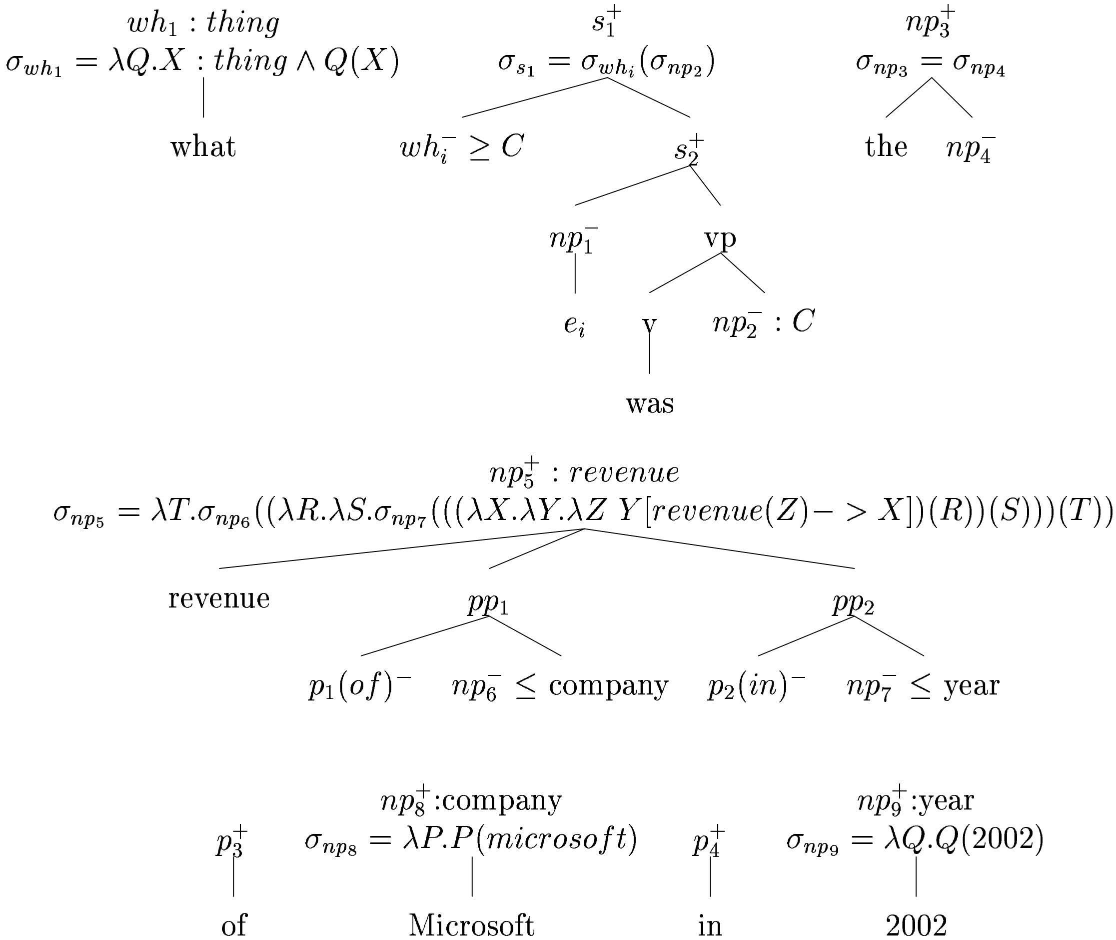

Fig. 3. Elementary trees for *What was the revenue of Microsoft in 2002?*

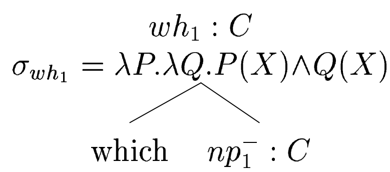

Fig. 4. Alternative elementary tree for *which*

The approach has been implemented as a parser operationalizing the calculus in [Muskens(2001)] and taking into account ontological information as described in [Cimiano and Reyle(2003)]. The parser has been successfully tested with the examples presented in this paper. In further experiments, we intend to integrate our approach with the F-logic inference engine OntoBroker [Decker et al.(1999)] and test the approach in a *in vivo* setting.
Also the coverage of the toy grammar presented in this paper will certainly need to be extended. Currently, we are developing a technique to automatically derive all the elementary trees from automatically learned subcategorization frames and corresponding selectional preferences/restriction as in

[Resnik(1997)]. In particular, our aim is to derive variants of the elementary trees via the application of certain rules. In addition, we intend to extend our analysis to handle multiple wh-questions, i.e. *Who owns what?* and temporal (*when*) questions as well as questions related to the reason *why* something happened.

Acknowledgments The work presented in this paper is partially supported by the IST-Dot.Kom project (http://www.dot-kom.org), sponsored by the EC as part of the framework V, (grant IST-2001-34038). The author would like to thank Steffen Staab as well as one of the anonymous reviewers of the workshop for their valuable comments.

References

[Androutsopoulos et al.(1995)] Androutsopoulos, I., Ritchie, G., Thanisch, P., 1995. Natural language interfaces to databases–an introduction. Journal of Language Engineering 1 (1), 29–81.

[Cimiano and Reyle(2003)] Cimiano, P., Reyle, U., 2003. Ontology-based semantic construction, underspecification and disambiguation. In: Proceedings of the Prospects and Advances in the Syntax-Semantic Interface Workshop.

[Decker et al.(1999)] Decker, S., Erdmann, M., Fensel, D., Studer, R., 1999. Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In: Meersman, R., et al. (Eds.), Database Semantics: Semantic Issues in Multimedia Systems. Kluwer, pp. 351–369.

[Gardent and Kallmeyer(2003)] Gardent, C., Kallmeyer, L., 2003. Semantic construction in feature-based tag. In: Proceedings of the 10th Meeting of the European Chapter of the Association for Computational Linguistics.

[Joshi and Schabes(1997)] Joshi, A., Schabes, Y., 1997. Tree-adjoining grammars. In: Handbook of Formal Languages. Vol. 3. Springer, pp. 69–124.

[Kifer et al.(1995)] Kifer, M., Lausen, G., Wu, J., May 1995. Logical foundations of object-oriented and frame-based languages. Journal of the Association for Computing Machinery .

[Montague(1974)] Montague, R., 1974. On the proper treatment of quantification in ordinary english. In: Thomason, R. H. (Ed.), Formal Philosophy: Selected Papers of Richard Montague. pp. 247–270.

[Muskens(2001)] Muskens, R., 2001. Talking about trees and truth-conditions. Journal of Logic, Language and Information 10 (4), 417–455.

[Resnik(1997)] Resnik, P., 1997. Selectional preference and sense disambiguation. In: Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?