Ontologies and Databases: myths and challenges

Enrico Franconi
Free University of Bozen-Bolzano, Italy
franconi@inf.unibz.it

ABSTRACT

After an introduction where the notion of ontology will be introduced in a rigorous way as a set of *constraints* over legal *database instances* playing the role of a *conceptual model* or of a set of *dependencies*, the tutorial will be divided in three parts. In the first part, we will discuss, mostly driven by examples, the role of ontologies in information systems design and the advantages and the challenges when adopting a formal approach based on logics. In the second part, a real-world tool for ontology design will be used to see the "logic" in action. In the third and most important part, we will discuss the use for information access of ontologies in data intensive scenarios based on database technologies, based on a scenario where a logic-based ontology mediates between the user information need and the data structured in the source database. The tutorial will emphasise the advantages of adopting a logic-based approach to the use of ontologies in data intensive applications, and the challenges that the research should still face to make this approach feasible and scalable in association with current database technology.

The audience can be of both database researchers and practitioners, since I will try to explain all the concepts through examples and the central demo, although the concepts that the audience will at the end get will be non-trivial ones. The main goal of this tutorial is to let the audience understand haw these novel technologies are non-trivial, but useful in perspective and worthwhile researching; I will also show that there are already some data intensive scenarios where a rigorous ontology-based approach is already applicable with success; at the same time, I will warn about the wrong usages of ontology-based technologies.

1. ONTOLOGIES AND DATABASES

Figure 1 graphically depicts the role of an ontology (the blue top layer) with respect to information structures (the middle layer) and to data (the bottom layer, modelled itself by the information structures). More precisely, an ontology is a logic-based conceptualisation of the information world,

PVLDB '08, August 23-28, 2008, Auckland, New Zealand
Copyright 2008 VLDB Endowment, ACM 978-1-60558-306-8/08/08

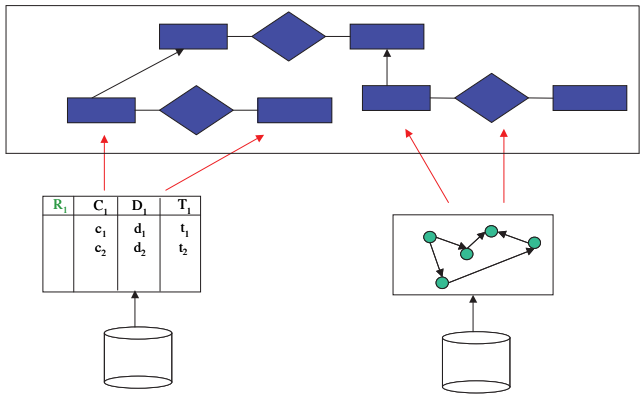

Figure 1: The general framework involving data, information, and knowledge.

specifying a set of *constraints* which declare what should necessarily hold for any data organised according to that information structure. In general, several kinds of information structures may be considered: relational databases, web pages, Java instances, XML semi-structured information, etc. In this framework, two major scenarios are conceivable.

The first obvious activity we can consider is the *conceptual modelling* activity [3]. Within this category – where scenarios considering ontologies as their ultimate object of analysis and use – we can analyse the life-cycle of the ontology itself: there is the creation of the ontology, the maintenance of the ontology, the deployment of the ontology as an object. The creation and the maintenance of an ontology may be conceived also in presence of proper data: for example, this would be the case of bootstrapping the design of an ontology from the available database schema, through a generalisation process. As a matter of fact ontology modelling deals with the question on how to describe in a declarative and reusable way the domain information of an application, its relevant vocabulary, and how to constrain the use the data, by understanding what can be drawn from it. The deployment of ontologies in *ontology-as-object* activities is typically an operation of navigation or intelligent querying of the ontology alone; this knowledge exploration activity is the kind of activity that is needed for example when classifying external documents or records.

In the tutorial I will adopt as a neutral language to represent ontologies *UML*-like class diagrams (see figure 2). The

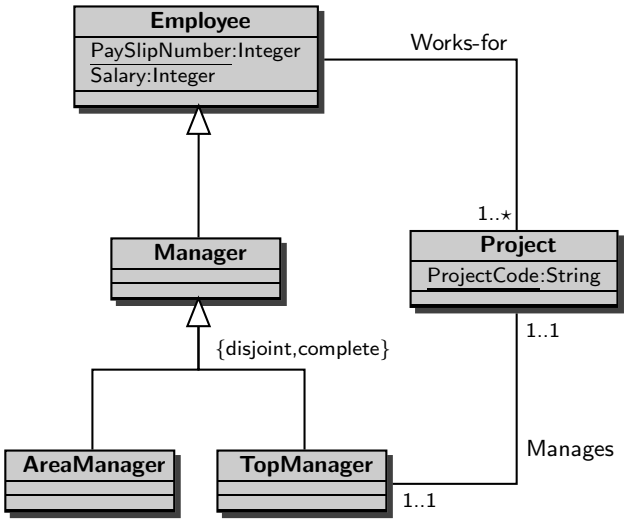

Figure 2: The UML diagram used as example.

advantage of this choice is that examples may be clear even to people who do not have familiarity with more classical ontology languages, due to its graphical nature and to the fact that UML class diagrams are quite widespread within computer technologies. I will show how non trivial tasks should be solved in the ontology-as-object scenario.

More challenging are the activities that involve an ontology together with its information source. Here, typical problems are the verification of the consistency of the data with respect to the knowledge, the reasoning involving both the query and the knowledge [5], and most notably the *access to the information* mediated by the ontology. Good ontologies put their emphasis on the correct and semantically rich representation of complex properties and relations that may exist in the data; they should allow for an abstract representation of information which resembles the way they are actually perceived and used in the real world, thus shortening (with respect to the more traditional data models) the semantic gap between the domain and its representation. With this perspective in mind, the user would prefer to query the information system using the richer vocabulary of the ontology. The vocabulary of the information structure could be seen in turn either as a subset of the conceptual vocabulary – this is the simplistic view – or more generally as a set of (materialised) views over the vocabulary of the ontology. However, in this case we have to solve the problem of view-based query processing [8, 7, 2]. The problem requires to answer a query posed to a database – the one defined by the ontology – only on the basis of the information in a set of (materialised) views, which are again queries over the same database. In the process, the information contained in the ontology should be of course taken into account. In this framework, more subtle problems come into play, since now the complexity of the scenario involves also the dimension of the data, which usually is much bigger than the dimension of the knowledge [4].

In the tutorial I will introduce the problem of accessing data mediated by an ontology at different degrees of complexity, from the simple cases to the more challenging ones; I will emphasise how those problems have been solved naively so far [6].

2. REFERENCES

[1] F. Baader, D. Calvanese, D. McGuinness, P. Patel-Schneider, and D. Nardi, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, Jan 2007.

[2] C. Beeri, A. Y. Levy, and M.-C. Rousset. Rewriting queries using views in description logics. In *PODS '97: Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 99–108, New York, NY, USA, 1997. ACM.

[3] A. Borgida, M. Lenzerini, and R. Rosati. Description logics for databases. In Baader et al. [1].

[4] D. Calvanese, G. De Giacomo, D. Lembo, and M. Lenzerini. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *Journal of Automated Reasoning*, 39(3):385–429, Oct 2007.

[5] D. Calvanese, G. D. Giacomo, and M. Lenzerini. On the decidability of query containment under constraints. In *Proc. of the 17th ACM SIGACT SIGMOD SIGART Sym. on Principles of Database Systems (PODS'98)*, pages 149–158, 1998.

[6] E. Franconi. Using ontologies. *IEEE Intelligent Systems*, 19(1), 2004.

[7] L. Segoufin and V. Vianu. Views and queries: determinacy and rewriting. *PODS '05: Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, Jun 2005.

[8] J. D. Ullman. Information integration using logical views. *Theoretical Computer Science*, 239(2):189–210, 2000.