Ontology-Based Data Management

Maurizio Lenzerini

Dipartimento di Ingegneria Informatica Automatica e
Gestionale Antonio Ruberti
Università di Roma La Sapienza, Roma, Italy
`http://www.dis.uniroma1.it/~lenzerini`

1 Introduction

While the amount of data stored in current information systems and the processes making use of such data continuously grow, turning these data into information, and governing both data and processes are still tremendously challenging tasks for Information Technology. The problem is complicated by the proliferation of data sources and services both within a single organization, and in cooperating environments.

The following factors contribute explaining why such a proliferation constitutes a major problem with respect to the goal of carrying out effective data governance tasks:

1. Although the initial design of a collection of data sources and services might be adequate, corrective maintenance actions tend to re-shape them into a form that often diverges from the original conceptual structure.
2. It is common practice to change a data source (e.g., a database) so as to adapt it both to specific application-dependent needs, and to new requirements. The result is that data sources often become data structures coupled to a specific application (or, a class of applications), rather than application-independent databases.
3. The data stored in different sources and the processes operating over them tend to be redundant, and mutually inconsistent, mainly because of the lack of central, coherent and unified coordination of data management tasks.

The result is that information systems of medium and large organizations are typically structured according to a "sylos"-based architecture, constituted by several, independent, and distributed data sources, each one serving a specific application. This poses great difficulties with respect to the goal of accessing data in a unified and coherent way. Analogously, processes relevant to the organizations are often hidden in software applications, and a formal, up-to-date description of what they do on the data and how they are related with other processes is often missing.

The introduction of service-oriented architectures is not a solution to this problem per se, because the fact that data and processes are packed into services is not sufficient for making the meaning of data and processes explicit. Indeed, services become other artifacts to document and maintain, adding complexity to the governance problem. Analogously, data warehousing techniques and the separation they advocate between the management of data for the operation level, and data for the decision level, do not provide solutions to this challenge. On the contrary, they also add complexity to the system, by replicating data in different layers of the system, and introducing synchronization processes across layers.

2 The notion of "Ontology-based data management"

All the above observations show that a unified access to data and an effective governance of processes and services are extremely difficult goals to achieve in modern information systems. Yet, both are crucial objectives for getting useful information out of the data stored in the information system, as well as for taking decisions based on them. This explains why organizations spend a great deal of time and money for the understanding, the governance, the curation, and the integration of data stored in different sources, and of the processes/services that operate on them, and why this problem is often cited as a key and costly Information Technology challenge faced by medium and large organizations today [4].

We argue that ontology-based data management (OBDM) is a promising direction for addressing the above challenges. The key idea of OBDM is to resort to a three-level architecture, constituted by the ontology, the sources, and the mapping between the two, where the ontology is a formal description of the domain of interest, and is the heart of the whole system. In this sense, OBDM can be seen as a form of information integration, where the usual global schema is replaced by the conceptual model of the application domain, formulated as an ontology expressed in a logic-based language. With this approach, the integrated view that the system provides to information consumers is not merely a data structure accommodating the various data at the sources, but a semantically rich description of the relevant concepts in the domain of interest, as well as the relationships between such concepts.

The distinction between the ontology and the data sources reflects the separation between the conceptual level, the one presented to the client, and the logical/physical level of the information system, the one stored in the sources, with the mapping acting as the reconciling structure between the two levels [9]. This sepration brings several potential advantages:

- The ontology layer in the architecture is the obvious mean for pursuing a declarative approach to information integration, and, more generally, to data governance. By making the representation of the domain explicit, we gain re-usability of the acquired knowledge, which is not achieved when the global schema is simply a unified description of the underlying data sources [6].
- The mapping layer explicitly specifies the relationships between the domain concepts on the one hand and the data sources on the other hand. Such a mapping is not only used for the operation of the information system, but also for documentation purposes. The importance of this aspect clearly emerges when looking at large organisations where the information about

data is widespread into separate pieces of documentation that are often difficult to access and rarely conforming to common standards. The ontology and the corresponding mappings to the data sources provide a common ground for the documentation of all the data in the organisation, with obvious advantages for the governance and the managament of the information system.
- A third advantage has to do with the extensibility of the system. One criticism that is often raised to data integration is that it requires merging and integrating the source data in advance, and this merging process can be very costly. However, the ontology-based approach does not impose to fully integrate the data sources at once. Rather, after building even a rough skeleton of the domain model, one can incrementally add new data sources or new elements therein, when they become available, or when needed, thus amortising the cost of integration. Therefore, the overall design can be regarded as the incremental process of understanding and representing the domain, the available data sources, and the relationships between them. The goal is to support the evolution of both the ontology and the mappings in such a way that the system continues to operate while evolving, along the lines of ”pay-as-you-go” data integration pursed in the research on data-spaces [12].

3 Challenges

OBDM is a new paradigm, which provides several interesting features. Many of them have been already proved effective in managing complex information systems. On the other hand, several important issues remain open, and constitute stimulating challenges for the research community. The following is a list of some of them.

- In OBDM, the client of the information system can interact with the system by means of an abstract representation of the domain. She can ask queries on the basis of the concepts of the domain, rather than the structures of the data sources. By taking into account the ontology and the mappings to the data sources, the OBDM system is in charge of translating the original query into a query to be evaluated at the source. How to do this translation correctly and efficiently is a fascinating research problem. A promising approach is the one based on rewriting [10], but we have to take into account that whather a rewriting approach is sound and complete depends on the expressive power of both the language used to express the ontology [8, 7], and the language to express the mapping. Also, how to extend query answering to cover the case of different information system architectures [5], or different data models [11] is still an issue to be deeply investigated.
- Since the ontology should reflect the conceptual model of the domain, and not the information at the sources, it is likely that data at the sources are not fully coherent with the axioms in the ontology. How to design incosistency tolerant query answering methods is one of the most important challenges in OBDM. It is interesting to note that this issue is very much related to

consistent query answering, which has been studied in databases in the last years [1].

- Although in classical data integration the main service to be delivered by the system is query answering, the OBDM should also provide the client with other functionalities. One important functionality is the update, that should be offered as a service [3] in an OBDM system. Analogously to queries, in OBDM, updates should be expressed at the level of the ontology, and this poses two main challenges. The first challenge is to define a convincing semantics of the update operations [13]. The second challenge is to design the OBDM system in such a way that the update request is translated into appropriate updates on the source data. How to do this translation is largely open, and has strong connections to the view-update problem in database [2].

References

1. Marcelo Arenas, Leopoldo E. Bertossi, Jan Chomicki. Consistent Query Answers in Inconsistent Databases. In *Proc. of ACM PODS 1999*, 68–79, 1999.
2. François Bancilhon, Nicolas Spyratos. Update Semantics of Relational Views. *ACM Transactions on Databases Systems*, 6(4), 557–575, 1981.
3. Daniela Berardi, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, Massimo Mecella. A foundational vision of e-services *Web Services, E-Business, and the Semantic Web* Springer Berlin Heidelberg, 28–40, 2004.
4. Philip A. Bernstein, Laura Haas. Information integration in the enterprise. *Comm. of the ACM*, 51(9):72–79, 2008.
5. Diego Calvanese, Elio Damaggio, Giuseppe De Giacomo, Maurizio Lenzerini, Riccardo Rosati Semantic data integration in P2P systems. *Databases, Information Systems, and Peer-to-Peer Computing*, Springer Berlin Heidelberg, 77–90, 2004.
6. Andrea Calì, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini. Accessing Data Integration Systems through Conceptual Schemas. *Proc. of ER'01*, 270–284, 2001, Spinger-Verlag, LNCS, Vol. 2224.
7. Andrea Calì, Georg Gottlob, Thomas Lukasiewicz. A General Datalog-Based Framework for Tractable Query Answering over Ontologies. In *Proc. of ACM PODS 1999*, 77–86, 1999.
8. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. of Automated Reasoning*, 39(3):385–429, 2007.
9. Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini. Answering queries using views in description logics. In *Proc. of KRDB'99*, volume 21 of *CEUR, ceur-ws.org*, 1999.
10. Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, Moshe Y. Vardi. View-based Query Processing: On the Relationship between Rewriting, Answering and Losslessness. *Theoretical Computer Science*, 371(3):169–182, 2007.
11. Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, Moshe Y. Vardi. View-Based Query Processing for Regular Path Queries with Inverse. In *Proc. of ACM PODS 2000*, 58-66, 2000.
12. Anish Das Sarma, Xin Dong, Alon Y. Halevy. Bootstrapping pay-as-you-go data integration systems. In *Proc. of ACM SIGMOD 2008*, pages 861–874, 2008.
13. H. Liu, C. Lutz, M. Milicic, F. Wolter. Updating Description Logic ABoxes. In *Proc. of KR 2006*, pages 46–56, 2006.