Data Integration in a System with Agents' Models*

Grażyna Brzykcy

Poznań University of Technology, Institute of Control and Information Engineering,
Pl. M. Skłodowskiej-Curie 5, 60-965 Poznań, Poland
Grazyna.Brzykcy@put.poznan.pl

Abstract. A problem of data integration in an environment consisting of a community of peer-to-peer cooperative agents is discussed. Data in the system are locally structured by the agents under different schemas. Communication and cooperation in the system are based on asking and answering queries by the agents. We show how to represent data and to merge related information from multiple data sources by means of unification operation and diverse Prolog mechanisms. The declarative approach is suitable to model data integration task and other constraints on data: from ontological to locally induced by an agent.

Keywords: P2P system, XML data, schema mapping, query rewriting, unification, Prolog-like computations.

1 Introduction

Distributed and heterogeneous nature of today's software systems is a reason for a lot of efforts in the domain of data integration [8,13,14,15,17,18]. In the paper a data integration problem is considered in systems consisted of autonomous components (agents, peers) [16] each of which manages some part of data (knowledge). An agent can independently decide how to structure its local data and can cooperate with other agents by asking them and answering queries. Peers decide when to join and leave the system, when to communicate and share their data with other partners. The SIXP2P system [6] for semantic integration of XML data, currently under development in Poznań University of Technology, is a representative example of such systems and forms a suitable testing ground for diverse solutions of data integration problem.

A P2P collection of data sources may be used as a general model of information structure. A query defined by a user and directed to a local agent in the structure is cooperatively answered by all the related peers in the P2P system. Partial answers from diverse sources are integrated and the result is presented by the local agent to the user in a form described in the query. How to model the agent's main action of integration of partial data?

We decided to take advantage of unification operation as the natural model of data integration. We also find it rational to use other Prolog-like mechanisms in the system. Logic programming environments, particularly those with types and extended

* The work was supported in part by the Polish Ministry of Science and Higher Education under Grant N516 015 31/1553.

N.T. Nguyen et al. (Eds.): KES-AMSTA 2008, LNAI 4953, pp. 162–171, 2008.

records (ψ-terms, feature structures) [1,2,9], provide not only a high level of abstraction for knowledge representation, but also a pattern matching, unification and subsumption – techniques used in data integration tasks. Moreover, the systems come with metaprogramming facilities, a backtracking search, executable specifications and diverse extensions (e.g., *LogicPeer*, *LogicWeb* [11], situations [12], constraint programming [1]). We have already prepared the declarative specification of semantic-driven communication between agents and between agents and the broker in the SIXP2P system [6] and have shown how to cover context in these rules [5]. Now the focus will be on a problem of representation of data, schemas and mappings between XML data and on the tasks of asking and answering queries, data exchange and data merging. Particularly, in the paper we address the following issues: how to represent data, schemas and mappings (section 3), how to represent queries and answers (section 4), how to perform data exchange task (section 5) and how to integrate data from different sources (section 5). In section 2 we shortly characterize the SIXP2P system.

As our ideas and solutions grow out of the analysis of Yu's and Popa's work [17] on query rewriting and data integration, we have decided to follow their line of presentation.

Declarative approach means that instead of imperative algorithms (e.g., defined in Java [17]) we use logical variables to denote entities (data), Prolog-like clauses to specify algorithms and logic programming systems to perform computations. Benefits of the approach are as follows:

1. Logical variables are suitable for data representation: to express data equality, to represent partial (partially instantiated) data, to achieve more readable specifications of mappings and queries.
2. Unification operation is a suitable model of data integration task and is automatically performed in Prolog systems. We are able to match data effectively.
3. Data merging in Prolog systems is performed with built-in terms (lists) and procedures (e.g., *member*).

2 Agents and Tasks in the SIXP2P System

The SIXP2P system for semantic integration of XML data consists of autonomous agents, which manage local data sources. An agent decides how to structure its data and when to share them with other partners. The data structure is described by a peer's local schema, which contains all the information that is vital to an agent to ask and answer queries. Therefore, a schema forms a model of an agent in the system [7].

Communication between agents is defined locally as each agent in the system can "see" only a subset of other agents. However, the "seeing" relation is transitive and induces a significant extension of the set of available data sources. Thus, cooperative query evaluation is possible in the system and the task is performed together with agents indirectly connected to the enquirer.

The main goal in the system consists in answering queries by taking advantage of all the knowledge sources that store semantically related data. There are two basic forms of achieving this goal: by data exchange and by data integration.