Managing Data Mappings in the Hyperion Project *

Anastasios Kementsietsidis Marcelo Arenas Renée J. Miller
Department of Computer Science, University of Toronto
{tasos, marenas, miller}@cs.toronto.edu

Abstract

We consider the problem of mapping data in peer-to-peer systems. Such systems rely on simple value searches to locate data of interest. However, different peers may use different values to identify or describe the same data. To accommodate this, peer-to-peer systems often rely on mapping tables that list pairs of corresponding values for search domains that are used in different peers. We illustrate how such tables are used in the Genomics community by expert curators. We then argue why mapping tables are appropriate for data mapping in a peer-to-peer environment and motivate the problem of managing these tables. The work presented here is part of the Hyperion Project [4].

1 Introduction

Traditionally, in multi-database systems, data integration and exchange between heterogeneous data sources is provided mainly through the use of views (logical translation programs) that map and restructure data between heterogeneous schemas [6, 10]. These programs or queries depend closely on the logical structures of the underlying data sources. To correctly restructure and map data, the sources must be willing to share at least portions of their schemas and cooperate in establishing and managing the translation programs and queries. In our work, we consider peer-to-peer settings (and more general networked applications) in which such close cooperation is either not desirable (perhaps for privacy reasons) or not feasible (perhaps due to resource limitations or the dynamic nature of the data structures) [8, 5].

To find data when there is little, or no agreement on the logical design of the data (or on how different logical designs correspond), we must focus on data values and how values correspond. If we can map values, particularly identifying values (names or keys), we can still request and exchange specific data of interest. This approach proves extremely useful in domains where there is no accepted naming standard. In such a setting, different peers may necessarily have had to develop their own naming conventions. Standards often emerge only after many heterogeneous sources have set up their own naming conventions. There may be many applications that depend on the use of the internal conventions. So, migration to conform to external standards is time-consuming and expensive. To find data in such environments, people have made use of *mapping tables* that store the correspondence between values. At their simplest, these tables are binary tables containing pairs of corresponding identifiers from two different sources. Such tables can be used in simple value searches where, for example, a peer who wishes to find a file called X it first consults a (shared or local) mapping table to find the name of X in the other peer. In general, we may need to map values containing multiple attributes (including both identifiers and descriptive attributes). For example, geographic locations may be indicated by pairs of longitude and latitude values in one peer and by some form of federal postal code in a second. However, we can still use these mapping tables to exchange data related to specific values. The query "retrieve all information related to postal code X" in peer one becomes "retrieve all information related to coordinates (Y, Z)" in peer two.

Mapping tables represent *expert knowledge* and are typically created by domain specialists. Indeed, currently the creation of mapping tables is a time-consuming and manual process performed by a set of expert curators. While widely used, especially in the biological domain [7], we are aware of no data management tools currently designed to facilitate the creation, maintenance and management of these tables.

2 Motivating Example

Consider an example drawn from the domain of biological databases. Currently, there is an overwhelming number of Genomic data sources ranging from large

*This research was supported by a grant from NSERC.

Proceedings of the 19th International Conference on Data Engineering (ICDE'03)
1063-6382/03 $ 17.00 © 2003 IEEE

public sources, such as GDB [1], to sources that are specific to individual research labs. Integration of these sources to provide uniform access for scientists, although extremely desirable, seems unattainable due to a myriad of political, financial and technical reasons [7]. Among the technical reasons is the inherent heterogeneity of the sources which range from relational databases to formatted files or spreadsheets. In addition, the schemas and formats of the sources evolve rapidly in response to new biological techniques and requirements.

To achieve some degree of integration, biologists commonly use what we have called *mapping tables*. A mapping table represents expert knowledge about a pair of related domains and is typically constructed manually by curators [7]. It is used between pairs of sources to associate data values that reside in the sources. For example, in the world of biological data sources, mapping tables can be used to relate gene data in one source to the related protein data in another source (where the gene is said to *encode for* the protein). Note that the mapping table is not necessarily a function, there may be many proteins related to a gene. Even a mapping table relating gene identifiers, may be many-to-many. This occurs often in biological sources where there may be aliases for the same identifier. As identifiers are updated, old identifiers may need to be kept if, for example, they refer to the content of static sources such as journal articles which may contain antiquated names and identifiers for entities.

In this poster, we motivate the main characteristics and uses of mapping tables. First, we show that mapping tables can be used to associate values not only within a single domain but across disparate domains. Second, we show that mapping tables are an appropriate tool to use in peer-to-peer systems since they respect the autonomy of the peers. Finally, we present some examples that motivate why reasoning capabilities are desirable in an environment were mapping tables are used.

Associations within and across domains: Notice that by using mapping tables we are able to associate seemingly unconnected databases, something that has been called *mediation across multiple worlds* [9]. In a typical integration scenario, we are often dealing with *one world*, for example, a set of sources all containing information about genes. However, there are situations where sources from *disjoint worlds* can be associated since the corresponding worlds are semantically close to each other. As an example, consider the gene Database (GDB) [1] and the SwissProt database [2]. The GDB database, apart from storing gene-related information, also has a mapping table in which it stores associations between gene identifiers from GDB and protein identifiers from SwissProt. An example of such a table is shown in Figure 1 (a). Table 2 of this figure associates genes and proteins with identifiers for genetic disorders represented in the MIM database [3]. Table 3 directly associates genes and genetic disorders. Each table may have been constructed by different curators with different (possibly overlapping) knowledge of the underlying domains.

Peer autonomy: Autonomy is of utmost importance in any peer-to-peer system and in many types of networked applications. Mapping tables respect the autonomy of the sources that they associate. To see this, notice that the mapping table shown in Figure 1 (a) does not express how genes and proteins are related in general, nor how they should be represented or stored in their respective sources. Rather, it only encodes the fact that a domain expert has determined that certain genes are related to certain proteins. Such information is necessary to effectively perform searches across peers.

Automated discovery of valid associations: In general, a mapping table consists of two disjoint sets of attributes X and Y (we use a double line in figures to distinguish between the two). A tuple (x, y) in the mapping table indicates that the value x is associated with y. Thus, a mapping table specifies valid associations between values in two peers. A set of mapping tables specify associations over a network of peers. In our work, we provide a small set of natural rules that curators may use to specify how sets of mapping tables may be combined. These rules allow curators to declare the extent of their knowledge. Our tools then automatically find new associations and identify inconsistencies in the tables.

Consider a single mapping table that associates values of X with values of Y. This table may represent complete knowledge of the domain X or only partial knowledge. In the former case, values X may only be associated with Y values if they are present in the table (and then only to the indicated Y values). Hence, values of X that do not appear in the mapping table cannot be associated with any values of Y. We call this a *closed-world* semantics. Alternatively, a curator who has only partial knowledge of a domain may specify an *open-world* semantics. An open-world table does not constrain how values of X that are not present in the table may be mapped. Hence, they may be associated with any value of Y. Under the open-world semantics, in Table 1 of Figure 1 (a), a gene that is not mentioned in the mapping table can be associated with any protein. This semantics recognizes that curators are often experts only on a subset of a domain. This is a semantics we found used often in environments where new data may be emerging dynamically. In more heavily curated domains, curators did wish to express complete knowledge about all values of X.

Proceedings of the 19th International Conference on Data Engineering (ICDE'03)
1063-6382/03 $ 17.00 © 2003 IEEE

GDB_id	SwissProt
g_1	p_2
g_2	p_4

(a) Mapping Table 1

GDB_id	SwissProt	MIM_id
g_1	p_1	m_1
g_1	p_2	m_2
g_2	p_3	m_3

(b) Mapping Table 2

GDB_id	MIM_id
g_3	m_4

(c) Mapping Table 3

Figure 1. An initial set of mapping tables.

Given a semantics for mapping tables, the simplest rule for combining them is to take their conjunction, i.e., to look for all the associations that satisfy all mappings. Consider the mapping tables shown in Figure 1. Recall that Table 2 indicates specific pairs of genes and proteins that can together be associated with a genetic disorder, while Table 3 associates genes directly with genetic disorders. Suppose curators have constructed these tables and used an open-world semantics for all three. Users may use Table 3 directly in their queries to associate genes with genetic disorders. However, they may wish to make use of Tables 1 and 2 (which were perhaps constructed by other curators) to obtain additional associations of genes with disorders. Under an open-world semantics, the association (g_1, m_2) can be derived from the mappings since we can find a *witness* tuple that involves all the attributes in the mappings, has g_1 as GDB_id and m_2 as MIM_id, and satisfies all the mappings. This tuple is $t = (g_1, p_2, m_2)$. Notice that t satisfies Table 1 since (g_1, p_2) is in this table and it satisfies Table 3 since g_1 is not mentioned in this table. Observe that (g_1, m_1) is not a valid association with respect to the mapping tables in the figure, since there is no witness tuple for these values (no value of SwissProt satisfies the conditions mentioned above). If one or more of the tables in Figure 1 have a closed-world semantics, the set of complete associations between GDB and MIM changes. In our work, we have developed algorithms for inferring a complete set of associations (aliases) and for determining if a set of mapping tables are inconsistent. Hence, curators can build mapping tables independently (autonomously) and yet make use of the knowledge of other curators at query time.

3 The Hyperion Project

The objective of the Hyperion project [4] is to investigate the data management issues that are raised in a peer-to-peer environment where each peer may have data to share with other peers. The main goals of the project are: the definition of a peer-to-peer data management architecture; the study of viable data integration, exchange, and mapping mechanisms; the development of algorithms for the efficient search, retrieval and exchange of data among peers. Mapping tables provide the foundation for exchanging information between peers. Our work ensures these tables can be built autonomously and yet used effectively in combination across a network of peers.

References

[1] Human Genome Database. http://www.gdb.org/.

[2] The SWISS-PROT Protein Knowledgebase. http://www.ebi.ac.uk/swissprot/.

[3] Online Mendelian Inheritance in Man. http://www.ncbi.nlm.nih.gov/omim/.

[4] The Hyperion Project. http://www.cs.toronto.edu/db/p2p/The_Hyperion_Project.html

[5] P. Bernstein, F. Giunchiglia, A. Kementsietsidis, J. Mylopoulos, L. Serafini and I. Zaihrayeu. Data Management for Peer-to-Peer Computing: A Vision. In WebDB'02.

[6] C-C. K. Chang and H. Garcia-Molina. Mind your Vocabulary: Query Mapping Across Heterogeneous Information Sources. In SIGMOD'99, pp. 335-346.

[7] S. Davidson, G. C. Overton, and P. Buneman. Challenges in Integrating Biological Data Sources. *Journal of Computational Biology* 2(4):557:572, 1995.

[8] S. Gribble, A. Halevy, Z. Ives, M. Rodrig, and D. Suciu. What can databases do for peer-to-peer? In WebDB'01.

[9] B. Ludäscher, A. Gupta and M. E. Martone. Model-based Mediation with Domain Maps. In ICDE'01., pp. 81-90.

[10] L. Popa, Y. Velegrakis, R. J. Miller, M. A. Hernandez, and R. Fagin. Translating Web Data. In VLDB'02, pp. 598–609.

Proceedings of the 19th International Conference on Data Engineering (ICDE'03)
1063-6382/03 $ 17.00 © 2003 IEEE