INTEROPERABILITY IN DATA WAREHOUSES

Riccardo Torlone
Roma Tre University
http://torlone.dia.uniroma3.it/

SYNONYMS

Data warehouse integration

DEFINITION

The term refers to the ability of combining the content of two or more heterogeneous data warehouses, for the purpose of cross-analysis. This need emerges in a variety of practical situations. For instance, when different designers of a large company develop their data marts independently, or when different organizations involved in the same project need to integrate their data warehouses.

Data Warehouse interoperability is a special case of the general problem of database integration, but it can be tackled in a more systematic way because data warehouses are structured in a rather uniform way, along the widely accepted concepts of dimension and fact. As it happens in the general case, different degrees of interoperability can be pursued by adopting standards and/or by applying reconciliation techniques, likely specific for this context. The problem is becoming increasingly relevant with the spreading of federated architectures. Nevertheless, it has been the focus of a few systematic works and numerous open problems remain to be solved.

HISTORICAL BACKGROUND

In spite of its relevance, the problem of data warehouse integration has received little attention so far. Conversely, the general problem of databases integration has been studied in the literature extensively and several aspects, both at scheme and instance level, have been deeply investigated, such as the automatic matching of terms and the resolution of structural conflicts (see [8, 12] for surveys on these topics).

In the specific context of data warehouses, Kimball [7] has identified the problem for the first time: he has investigated the integration of heterogeneous dimensions in a scenario of data warehouse design and has introduced the informal notions of dimension *conformity*. Intuitively, two dimensions are conformed if their share some information in a consistent way. This is an important requirement in *drill-across queries*, which are basically joins of different facts over common dimensions. The notion of conformity has been formalized and extended by Cabibbo and Torlone in the context of data mart integration [3] under the name of dimension *compatibility*: they have demonstrated that this property gives the ability to perform correct drill-across queries over heterogeneous data marts.

An issue related to the integration of data warehouses, which has been studied in the context of statistical databases, is the *derivability* of summary data. This notion has been defined by Sato [14] as the problem of deciding whether a summary data (which is, in a statistical database, the counterpart of a fact table) can be inferred from another summary data aggregated in a different way. The concept has been extended by Malvestuto [9], by considering the case in which the source is composed by several heterogeneous data sets: he proposes an algebraic approach to this problem and provides some necessary and sufficient conditions of derivability. Unfortunately, statistical databases have some similarity with multidimensional databases, but also some important diversities: this makes the application of these approaches to data warehouses not easy.

Some related work has been done on the problem of integrating a data warehouse with external data stored in XML [6] and in object-oriented [11] format, but just a few works have been devoted to the specific problem of the interoperability between heterogeneous data warehouses. They will be discussed in the following section.

While current commercial tools do not provide a complete support for data warehouse interoperability, they offer facilities that can be very useful in this framework, such as metadata import/export (using XML) and standardized ways to represent data (using a multidimensional model).

SCIENTIFIC FUNDAMENTALS

Since data warehouse interoperability can be considered a special case of the problem of database integration, general data reconciliation techniques can be often used. For instance, methods for the automatic matching of terms or for the resolution of structural conflicts. In addition, it is possible to take advantage on the fact that, in this context, the data sources always have a multidimensional structure. Therefore, the problem can be addressed by focusing on the reconciliation of heterogeneous dimensions and facts. Following this observation, the section discusses: standards that can be adopted to support data warehouse interoperability, conflicts that can arise in this context, and methodologies that can be used to perform the integration.

1 Standards

An important support for interoperability can be provided by the adoption of standards. Initially, two industry standards have been proposed by multi-vendor organizations for data warehouses: the Open Information Model (OIM) developed by the Meta Data Coalition (MDC), and the Common Warehouse Metamodel (CWM) developed by the Object Management Group (OMG) [16]. Later, MDC and OMG joined their efforts and proposed a new version of the CWM as the standard metadata model. The Common Warehouse Metamodel is a platform-independent specification for exchanging multidimensional data between different platforms and tools. It is based on the standards UML, XMI, and MOF, and provides a set of generic, external representations of metadata, called *metamodels*, that provide a comprehensive framework for data exchange. These metamodels can be used to describe the various components of the data warehouse architecture: data sources, ETL processes, multidimensional cubes, relational tables, and so on. However, it has been observed that their expressivity is not sufficient to capture all the complex semantics of conceptual multidimensional models, so they hardly can be used for effective integration of different data warehouses [13].

2 Conflicts

In the integration of different multidimensional data sources, a number of conflicts can arise, both at the schema and at the instance level.

- *Dimension conflicts*:
  - Schema: conflicts can arise on entity names (e.g., different names for the same dimensions and/or different names for similar levels of two dimensions) and on dimension hierarchies (similar dimensions organized over different levels of aggregation and/or inconsistencies on the roll-up relationships between levels).
  - Instance: conflicts can arise on member names (different names for the same members of different dimensions) and on the members of dimensions (similar dimensions populated by different members).
- *Fact conflicts*
  - Schema: still, conflicts can arise on names (different names for the same measures) and on dimensions that differ in number and/or in the levels of aggregation.
  - Instance: conflicts can arise on measures (inconsistent values for the same measures and/or differences in scales).

As mentioned in the previous section, Cabibbo and Torlone [3] have identified a fundamental property that should be enforced while solving conflicts between heterogeneous data warehouses: dimension and fact *compatibility.* Two different dimensions d_1 and d_2 are compatible when their common information is consistent, that is, when aggregations computed over d_1 and d_2 and aggregations computed over the dimension obtained by merging d_1 and d_2 produce the same results. Having compatible dimensions and facts is important because it gives the ability to look consistently at data across data marts and to combine and correlate such data by means of drill across queries. Building on this notion, they have also identified a number of desirable properties that a *matching* between dimensions (that is, a correspondence between their levels) should satisfy: (i) the *coherence* of the hierarchies on levels, (ii) the *soundness* of the levels in correspondence, according to the members associated with them, and (iii) the *consistency* of the roll-up functions that relate members of different levels within the matched dimensions.

3 Integration techniques

Two heterogeneous data warehouses can be combined if they share one or more dimensions and can be actually integrated if their facts can be joined, in a consistent way, over such common dimensions. It follows that a general methodology for achieving interoperability in data warehouses includes the following steps:

1. identification of the facts that can be integrated and the dimensions of these facts that can be combined to perform the integration,
2. resolution of conflicts between common dimensions,
3. resolution of conflicts between facts to be integrated,
4. reconciliation and integration of dimensions and facts according to the desired level of interoperability.

While this process can be supported by general reconciliation techniques based, for instance, on domain ontologies, it is possible to rely on specific techniques that take into account the rather standard structure of dimensions and facts. As usual, the level of interoperability can range from a scenario of loosely coupled integration, in which there is just the need to identify the common information between sources while preserving their autonomy, to a scenario of tightly coupled integration, in which the goal is rather merging the sources. In the former approach, queries are performed over a virtual view defined on the original sources, in the latter, queries are performed against a materialized view built from the sources.

M. Banek et al. [1] have addressed the problem of matching schema structures specific to data warehouses, the initial step of the above methodology. Their approach consists of two basic tasks. First, similarity matches between multidimensional structures are identified by comparing their names, data types and substructures (e.g., matches cannot violate the partial order in hierarchies). Then, heuristic rules, based on graph similarity, are used to choose the actual mappings, among the possible matches.

A methodology for the resolution of conflicts that guides the designers through the combination of independent data cubes has been proposed by Berger and Schrefl [2]. They also propose a specific language called SQL-MDi (SQL for multi-dimensional integration), supporting the methodology. In their approach, the goal is the generation of a tightly coupled architecture that combine heterogeneous multidimensional data sources into a materialized warehouse.

Cabibbo and Torlone [4] have proposed two practical approaches to the integration of autonomous data warehouses that try to enforce matchings satisfying the properties discussed in the previous section and refer to the scenarios of loosely and tightly coupled integration, respectively. As a preliminary tool, they introduce a powerful technique, the *chase of dimensions*, that can be used in both approaches to test for consistency and combine the content of the dimensions to integrate. This technique operates over a tableau populated by the members of the dimensions to be integrated, and makes use of the roll-up functions defined over such dimensions. Two integration algorithms are then proposed. The first algorithm provides the operations, expressed in an abstract algebra, that applied to the original dimensions, allow the specification of correct drill-across joins between the heterogeneous sources. The second algorithm generates new dimensions and facts, obtained by merging the original data sources, that constitute the reconciled data warehouse.

From a practical point of view, a general federated architecture supporting the interoperability of distributed and autonomous data warehouses has been proposed by Mangisengi et al. [10]. Tseng and Chen [15] have proposed a framework in which, after a resolution of conflicts, autonomous data cubes are first transformed into XML documents, then conflicts are solved by means of XQuery operations, and finally the access to integrated data is achieved through queries posed over an XML global view.

KEY APPLICATIONS

A common practice for building a data warehouse is to implement a series of data marts, each of which provides a dimensional view of a single business process [7]. These data marts should be based on common dimensions but what happens in practice is that, very often, different departments of the same company develop their data marts independently. It turns out that methods and tools for data warehouse reconciliation are very useful in such common situation.

Indeed, the need for combining autonomous data warehouse arises in other common scenarios. For instance, when different companies merge or get involved in a federated project or when there is the need to combine a proprietary data warehouse with data available elsewhere, for instance, in external and likely heterogeneous information sources, or in multidimensional data wrapped from the Web.

Furthermore, methods supporting data warehouse interoperability can be useful when there is the need to migrate a data mart from one implementation platform to another.

FUTURE DIRECTIONS

The area of data warehouse interoperability is largely unexplored and there is still a compelling need of systematic studies and effective tools. From a conceptual point of view, the problem needs a deeper investigation that takes into account, for instance, cases in which the structure of the data warehouses to be combined is non standard (e.g., for the presence of non-strict hierarchies or many-to-many relationships between facts and dimensions). In particular, the presence of irregular hierarchies makes the problem of dimension compatibility much harder since it requires complex tests at instance level. From a practical point of view, there is still a lack of effective tools specifically supporting the integration of autonomous and heterogeneous data warehouses.

EXPERIMENTAL RESULTS

Some preliminary tool supporting the interoperability of data warehouses has been recently proposed [5].

CROSS REFERENCE

Data warehousing systems: foundations and architectures. Multidimensional modeling. Logical data integration. Querying over data integration systems.

RECOMMENDED READING

[1] M. Banek, B. Vrdoljak, A. Min Tjoa, and Z. Skocir. Automating the Schema Matching Process for Heterogeneous Data Warehouses. In *Proc. of 9th Int. Conference on Data Warehousing and Knowledge Discovery, (DaWaK'07)*, pp. 45–54, 2007.

[2] S. Berger and M. Schrefl. Analysing Multi-dimensional Data Across Autonomous Data Warehouses. In *Proc. of 8th Int. Conference on Data Warehousing and Knowledge Discovery, (DaWaK'06)*, pp. 120–133, 2006.

[3] L. Cabibbo and R. Torlone. On the Integration of Autonomous Data Marts. In *Proc. of 16th Int. Conference on Scientific and Statistical Database Management (SSDBM'04)*, pp. 223–234, 2004.

[4] L. Cabibbo and R. Torlone. Integrating Heterogeneous Multidimensional Databases. In *Proc. of 17th Int. Conference on Scientific and Statistical Database Management (SSDBM'05)*, pag. 205–214, 2005.

[5] L. Cabibbo, I. Panella, and R. Torlone. DaWaII: a Tool for the Integration of Autonomous Data Marts. In *Proc. of 22nd Int. Conference on Data Engineering (ICDE'06), Demo session*, 2006.

[6] M.R. Jensen, T.H. Møller, and T.B. Pedersen. Specifying OLAP Cubes on XML Data. *Journal of Intell. Information Systems*, vol. 17, n. 2-3, pp. 255–280, 2001.

[7] R. Kimball and M. Ross. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling.* John Wiley & Sons, Second edition, 2002.

[8] M. Lenzerini. Data Integration: A Theoretical Perspective. In *Proc. of 21st Int. ACM Symp. on Principles of Database Systems (PODS'02)*, pp. 233–246, 2002.

[9] F. M. Malvestuto. The Classification Problem with Semantically Heterogeneous Data. In *Proc. of ACM SIGMOD Int. Conference on Management of Data*, pp. 157–176, 1988.

[10] O. Mangisengi, J. Huber, C. Hawel, and W. Eßmayr. A Framework for Supporting Interoperability of Data Warehouse Islands Using XML. In *Proc. of 3th Int. Conference on Data Warehousing and Knowledge Discovery, (DaWaK'01)*, pp. 328–338, 2001.

[11] T.B. Pedersen, A. Shoshani, J. Gu, and C.S. Jensen. Extending OLAP Querying to External Object Databases. In *Proc. of 9th Int. Conference on Information and Knowledge Management (CIKM 2000)*, pp. 405–413, 2000.

[12] E. Rahm and P.A. Bernstein. A Survey of Approaches to Automatic Schema Matching. *VLDB Journal*, vol. 10, n. 4, pp. 334-350, 2001.

[13] S. Rizzi, A. Abelló, J. Lechtenbörger, and J. Trujillo. Research in Data Warehouse Modeling and Design: Dead or Alive?. In *Proc. of the 9th ACM Int. Workshop on Data Warehousing and OLAP (DOLAP 2000)*, pp. 3–10, 2006 .

[14] H. Sato. Handling Summary Information in a Database: Derivability. In *Proc. of ACM SIGMOD International Conference on Management of Data*, pp. 98–107, 1981.

[15] F.S.C. Tseng and C.W. Chen. Integrating heterogeneous data warehouses using XML technologies *Journal of Information Science*, vol. 31, n.3, pp. 209–229, 2005.

[16] T. Vetterli, A. Vaduva, and M. Staudt. Metadata standards for data warehousing: Open Information Model vs. Common Warehouse Metamodel. *ACM SIGMOD Records*, vol. 29, n. 3, 2000.