New Frontiers in Business Intelligence: Distribution and Personalization

Stefano Rizzi*

DEIS - University of Bologna, V.le Risorgimento 2, 40136 Bologna, Italy

Abstract. To meet the new, more sophisticated needs of decision makers, a new generation of BI systems is emerging. In this paper we focus on two enabling technologies for this new generation, namely distribution and personalization. In particular, to support complex business scenarios where multiple partner companies cooperate towards a common goal, we outline a distributed architecture based on a network of collaborative, autonomous, and heterogeneous peers, each offering monitoring and decision support functionalities to the other peers. Then we discuss some issues related to OLAP query reformulation on peers, showing how it can be achieved using semantic mappings between the local multidimensional schemata of peers. Finally, as to personalization, we discuss the benefits of annotating OLAP queries with preferences, focusing in particular on how preferences enable peer heterogeneity in a distributed context to be overcome.

Keywords: Business Intelligence, Distributed Data Warehousing, User Preferences.

1 Introduction

Business intelligence (BI) transformed the role of computer science in companies from a technology for passively storing data into a discipline for timely detecting key business factors and effectively solving strategic decisional problems. However, in the current changeable and unpredictable market scenarios, the needs of decision makers are rapidly evolving as well. To meet the new, more sophisticated user needs, a new generation of BI systems (often labeled as *BI 2.0*) has been emerging during the last few years. Among the characterizing trends of these systems, we mention BI as a service, real-time BI, collaborative BI, and pervasive BI.

In this paper we focus on two enabling technologies for BI 2.0, namely distribution and personalization.

* Part of this work has been jointly carried out in collaboration with Matteo Golfarelli, Wilma Penzo, Elisa Turricchia (Univ. of Bologna, Italy), and Federica Mandreoli (Univ. of Modena and Reggio Emilia, Italy).

B. Catania, M. Ivanović, and B. Thanlheim (Eds.): ADBIS 2010, LNCS 6295, pp. 23–30, 2010.

1.1 Motivating Scenario and Envisioned Architecture

Cooperation is seen today by companies as one of the major means for increasing flexibility and innovating so as to survive in today uncertain and changing market. Companies need strategic information about the outer world, for instance about trading partners and related business areas. Indeed, it is estimated that above 80% of waste in inter-company and supply-chain processes is due to a lack of communication between the companies involved.

In such a complex and distributed business scenario, where multiple partner companies/organizations cooperate towards a common goal, traditional BI systems are no longer sufficient to maximize the effectiveness of monitoring and decision making processes. Two new significant requirements arise:

- Cross-organization monitoring and decision making: Accessing local information is no more enough, users need to transparently and uniformly access information scattered across several heterogeneous BI platforms [8].
- Pervasive and personalized access to information: Users require that information can be easily and timely accessed through devices with different computation and visualization capabilities, and with sophisticated and customizable presentations [14].

The architecture we envision to cope with this scenario is that of a dynamic, collaborative network of peers, each hosting a local, autonomous BI platform (see Figure 1). Each peer relies on a local multidimensional schema that represents the peer's view of the business and offers monitoring and decision support functionalities to the network users. Users transparently access business information distributed over the network in a pervasive and personalized fashion. Access is secure, depending on the access control and privacy policies adopted by each peer.

A typical user interaction in this context is the following. A user formulates an OLAP query q by accessing the local multidimensional schema of her peer, p. She

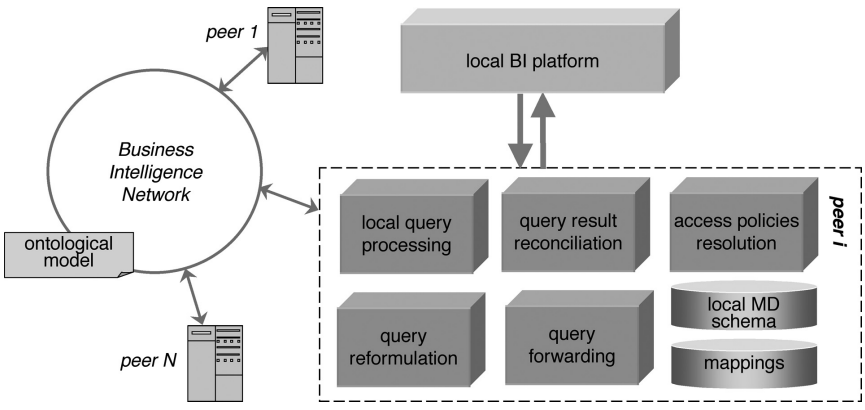

Fig. 1. Envisioned architecture for a collaborative BI network

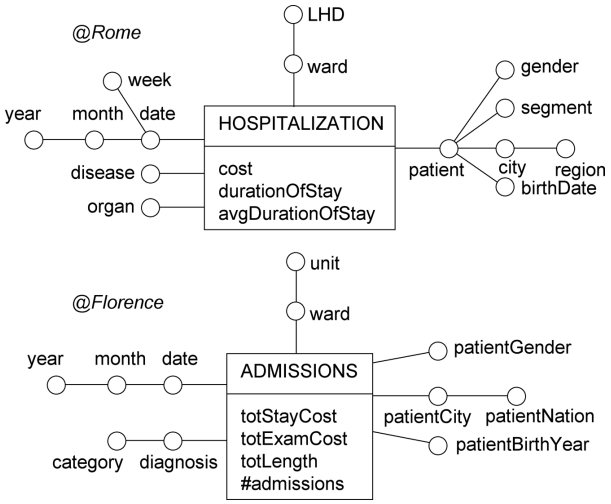

Fig. 2. Multidimensional schemata of related facts at two peers

can annotate q by a preference that enables her to rank the returned information according to her specific interests. To enhance the decision making process, q is forwarded to the network and reformulated on the other peers in terms of their own multidimensional schemata. Each involved peer locally processes the reformulated query and returns its (possibly partial or approximate) results to p. Finally, the results are integrated, ranked according to the preference expressed by the user, and returned to the user based on the lexicon used to formulate q.

In the working example we adopt in this paper, a set of local health-care departments participate in a collaborative network to integrate their data about admissions so as to enable more effective analysis of epidemics and health-care costs by the Ministry. For simplicity we will focus on two peers: the first, located in Rome, hosting data on hospitalizations at patient-level detail; the second, located in Florence, hosting data on admissions grouped by patient gender, residence city, and birth year. The underlying multidimensional schemata for these two peers are shown in Figure 2, using the Dimensional Fact Model notation [4].

2 Distribution

Although the integration of heterogeneous databases has been widely discussed in the literature, only a few works are specifically focused on strategies for data warehouse integration [2,16] and federation [1]. Indeed, in this context, problems related to data heterogeneity are usually solved by ETL (Extraction, Transformation, and Loading) processes that read data from several data sources and load them in a single multidimensional repository to be accessed by users. While this centralized architecture may fit the needs of old-style, stand-alone companies, it is hardly feasible in the context of a collaborative BI network, where the

dynamic nature of the business, together with the independence and autonomy of peers, call for more sophisticated solutions.

Peer Data Management Systems (PDMSs [7]) have been proposed in the literature as architectures to support sharing of operational data across networks of peers while guaranteeing peer autonomy, based on interlinked semantic mappings that mediate between the heterogeneous schemata exposed by peers [10]. The architecture we outlined in the previous section is in line with the PDMS infrastructure, but requires a number of specific issues —mostly related to the multidimensional nature of the information exchanged— to be faced:

- Query reformulation on peers is a challenging task due to the presence of aggregation and to the possibility of having information represented at different granularities and under different perspectives in each peer.
- The strategic nature of the exchanged information and its multidimensional structure, as well as the presence of participants that belong to different organizations, require advanced approaches for security, ranging from proper access policies to data sharing policies that depend on the degree of trust between participants, as well as techniques for protecting against undesired information inference.
- Mechanisms for controlling data provenance and quality in order to provide users with information they can rely on should be provided. A mechanism for data lineage is also necessary to help users understand the semantics of the retrieved data and how these data have been transformed to handle heterogeneity.
- A unified, integrated vision of the heterogeneous information collected must be returned to users. To this end, object fusion functionalities that take into account the peculiarities of multidimensional data must be adopted.

As a first step in this direction, we are currently working on query reformulation. In particular, we devised a language for the definition of semantic mappings between the multidimensional schemata of peers, and we introduced a query reformulation framework that relies on the translation of these mappings towards a ROLAP platform. A basic multidimensional model is considered, where a fact (e.g., patient admissions) is associated to a set of coordinates called dimensions (e.g., ward and admission date) and is quantified by a set of numerical measures (e.g., the total cost of the stay). A dimension can be further described by a set of hierarchically-structured attributes connected by many-to-one associations (e.g., a patient lives in one city, that in turn belongs to one region). As to the workload, we consider OLAP queries that can be expressed in GPSJ (Generalized Projection - Selection - Join) form [6]:

$$\pi_{G,AGG(m)}(\sigma_P(\chi))$$

where π denotes a generalized projection, i.e., an aggregation of measure m using aggregate function $AGG()$ over the attributes in G; σ_P is a selection based on Boolean predicate P; and χ denotes the star join between the fact table and the dimension tables. For instance, the following query expressed at the Rome peer

computes the total hospitalization cost of female patients for each region and year:

$$\pi_{region,year,SUM(cost)}(\sigma_{gender='F'}(\chi_{Rome}))$$

Now, let p and q be two peers in a collaborative BI network. The language we propose to express how the local multidimensional schemata of p maps onto the one of q includes five mapping predicates, namely same, equi-level, roll-up, drill-down, and related. Each mapping establishes a semantic relationship from a list of concepts (measures or attributes) of p (on the left side of the mapping predicate) to a list of concepts of q (on the right side of the mapping predicate), and enables a query formulated on p to be (exactly or approximately) reformulated on q. Optionally, a mapping can be associated with an encoding function that specifies how values of the left-side list of concepts can be obtained from values of the right-side list of concepts. If this function is available, it is used during query reformulation and data integration to return more query-compliant results to users.

While discussing in detail the whole set of mapping predicates and the query reformulation framework is outside the scope of this paper, we will give an intuition of the underlying mechanism with a basic example.

Example 1. The same predicate is used to state that a set c of measures in p has the same semantics than a set d of measures in q. If knowledge is available about how values of c can be derived from values of d, it can be expressed by two encoding functions $f : dom(c) \rightarrow \mathbb{R}$ and $g : dom(d) \rightarrow \mathbb{R}$. The semantics of these functions is that, whenever $f(c)$ is asked in a query at p, it can safely be rewritten as $g(d)$ at q. For instance,

$$< \mathsf{Rome.cost} > \mathsf{same}_{f,g} < \mathsf{Florence.totStayCost}, \mathsf{Florence.totExamCost} >$$

with

$$f(< \mathsf{cost} >) = \mathsf{cost}$$
$$g(< \mathsf{totStayCost}, \mathsf{totExamCost} >) = \mathsf{totStayCost} + \mathsf{totExamCost}$$

states that measure cost can be derived by summing totStayCost and totExamCost.

Similarly, the roll-up predicate states that a set c of attributes in p is a roll-up of (i.e., it aggregates) a set d of attributes in q. If knowledge is available about how to roll-up values of d to values of c, it can be expressed by a non-injective encoding function $f : dom(d) \rightarrow dom(c)$ that establishes a one-to-many relation between values of c and values of d, and is used to aggregate data returned by q and integrate them with data returned by p. For instance,

$$< \mathsf{Rome.week} > \mathsf{roll\text{-}up}_f < \mathsf{Florence.date} >$$

with $f(< \mathsf{date} >) =< \mathsf{week} : weekOf(\mathsf{date}) >$, states that weeks are an aggregation of dates.

Now consider the OLAP query (formulated at Rome) asking for the weekly hospitalization costs:

$$\pi_{\mathsf{week},SUM(\mathsf{cost})}(\chi_{Rome})$$

This query group-by is reformulated using the roll-up mapping from week to date, while measure cost is derived using the same mapping:

$$\pi_{weekOf(\mathsf{date}),SUM(\mathsf{totStayCost}+\mathsf{totExamCost})}(\chi_{Florence})$$

□

3 Personalization

Personalizing e-services by allowing users to express preferences is becoming more and more common. When querying, expressing preferences is seen as a natural way to avoid empty results on the one hand, information flooding on the other. Besides, preferences allow for ranking query results so that the user may first see the data that better match her tastes.

Though a lot of research has been carried out during the last few years on database preferences (e.g., [11,3]), the problem of developing a theory of preferences for multidimensional data has been mostly neglected so far with a few exceptions [15,9,17,12]. Indeed, expressing preferences could be valuable in this domain because:

- Preferences enable users to focus on the most interesting data. This is particularly beneficial in the OLAP context, since multidimensional databases store huge amounts of data. Besides, OLAP queries may easily return huge volumes of data (if their group-by sets are too fine), but they may return little or no information as well. The data ranking entailed by preferences solves both these problems.
- During an OLAP session, the user often does not exactly know what she is looking for. The reasons behind a specific phenomenon or trend may be hidden, and finding those reasons by manually applying different combinations of OLAP operators may be very frustrating. Preferences enable users to specify a "soft" pattern that describes the type of information she is searching for.
- In a collaborative BI network like the one envisioned in Section 1.1, heterogeneity in peers' multidimensional schemata and data may lead to obtaining empty results when reformulating queries, while a query annotated with a preference can produce meaningful results even when a common schema is not defined and the searched data are not found.

The last motivation plays a key role in the distributed framework outlined in this work. In our health-care example, consider a query asking in Rome for some statistics on hospitalizations, aggregated at the finest level along the patient hierarchy. If the patient group-by were expressed as a "hard" constraint, no data from Florence could be returned because the patient granularity is not present there (see Figure 2). If the patient group-by is expressed in a "soft" form using a preference, instead, data aggregated by patient gender, city, and birth year

can be returned from Florence. Though this granularity does not exactly match the one preferred by the user, the resulting information could be valuable in improving decision-making effectiveness. Similarly, a query in Rome could ask for data on hospitalizations whose duration of stay exceeds, say, 30 days. If the Florence peer stores no admissions yielding durations higher than 30, it can still return its admissions ranked by decreasing durations.

In [5] we presented MYOLAP, an approach for expressing and evaluating OLAP preferences. From the expressiveness point of view, the main features of our approach can be summarized as follows:

- Preferences can be expressed not only on attributes, that have categorical domains, but also on measures, that have numerical domains. Remarkably, the preference constructors that operate on attributes take the presence of hierarchies into account.
- Preferences can also be formulated on the aggregation level of data, i.e., on group-by sets, which comes down to expressing preferences on schema rather than on instances.
- Preferences can be freely composed using the Pareto and prioritization operators, thus forming an algebra that can easily be incorporated into a multidimensional query language like MDX [13].

We close this section by showing how the two preference query suggested above can be formulated in MYOLAP.

Example 2. The first query, asking for total hospitalization cost preferably aggregated by patient, can be annotated with the simple MYOLAP preference FINEST(PATIENT), stating that data should be preferably aggregated at the finest group-by set along the PATIENT hierarchy. While at Rome the finest group-by is patient, at Florence the finest group-by is patientCity, patientGender, patientBirthYear. Of course, proper visualization techniques will be required for displaying results at different granularities together while preserving the typical intuitiveness and navigability of OLAP interfaces.

The second query asks for hospitalizations whose duration of stay exceeds 30 days. To express the constraint about duration in a "soft" way, a MYOLAP preference like BETWEEN(durationOfStay,30,999) can be used. The best match for this preference are the hospitalizations whose duration of stay is higher than 30; however, if no such data are found, the returned hospitalizations are ranked in decreasing order by their durationOfStay values. □

4 Conclusions

In this paper we have outlined a peer-to-peer architecture for supporting distributed and collaborative decision-making scenarios. We have shown, using some examples, how an OLAP query formulated on one peer can be reformulated on a different peer, based on a set of inter-peer semantic mappings. The we have discussed the role of OLAP preferences in overcoming peer heterogeneity in both schemata and data.

Currently, we are finalizing a reformulation algorithm that relies on the translation of semantic mappings towards a ROLAP implementation. However, a large number of issues still need to be faced, as mentioned in Section 2. Our future work in this direction will be mainly focused on multidimensional-aware object fusion techniques for integrating data returned by different peers, on smart algorithms for routing queries to the most "promising" peers in the BI network, and on optimizing OLAP preferences in a distributed context.

References

1. Abiteboul, S.: Managing an XML warehouse in a P2P context. In: Eder, J., Missikoff, M. (eds.) CAiSE 2003. LNCS, vol. 2681, pp. 4–13. Springer, Heidelberg (2003)
2. Banek, M., Vrdoljak, B., Tjoa, A.M., Skocir, Z.: Automated integration of heterogeneous data warehouse schemas. IJDWM 4(4), 1–21 (2008)
3. Chomicki, J.: Preference formulas in relational queries. ACM TODS 28(4), 427–466 (2003)
4. Golfarelli, M., Rizzi, S.: Data Warehouse design: Modern principles and methodologies. McGraw-Hill, New York (2009)
5. Golfarelli, M., Rizzi, S., Biondi, P.: MYOLAP: An approach to express and evaluate olap preferences. IEEE Trans. Knowl. Data Eng. (to appear 2010)
6. Gupta, A., Harinarayan, V., Quass, D.: Aggregate-query processing in data warehousing environments. In: Proc. VLDB, Zurich, Switzerland, pp. 358–369 (1995)
7. Halevy, A.Y., Ives, Z.G., Madhavan, J., Mork, P., Suciu, D., Tatarinov, I.: The Piazza peer data management system. IEEE Trans. Knowl. Data Eng. 16(7), 787–798 (2004)
8. Hoang, T.A.D., Nguyen, B.: State of the art and emerging rule-driven perspectives towards service-based business process interoperability. In: Proc. Int. Conf. on Computing and Communication Technologies, Danang City, Vietnam, pp. 1–4 (2009)
9. Jerbi, H., Ravat, F., Teste, O., Zurfluh, G.: Applying recommendation technology in OLAP systems. In: Proc. ICEIS, Milan, Italy, pp. 220–233 (2009)
10. Kehlenbeck, M., Breitner, M.H.: Ontology-based exchange and immediate application of business calculation definitions for online analytical processing. In: Pedersen, T.B., Mohania, M.K., Tjoa, A.M. (eds.) DaWaK 2009. LNCS, vol. 5691, pp. 298–311. Springer, Heidelberg (2009)
11. Kießling, W.: Foundations of preferences in database systems. In: Proc. VLDB, Hong Kong, China, pp. 311–322 (2002)
12. Koutrika, G., Ioannidis, Y.: Answering queries based on preference hierarchies. In: Proc. VLDB, Auckland, New Zealand (2008)
13. Microsoft: MDX reference (2009), `http://msdn.microsoft.com/en-us/library/ms145506.aspx`
14. Rizzi, S.: OLAP preferences: a research agenda. In: Proc. DOLAP, Lisbon, Portugal, pp. 99–100 (2007)
15. Stefanidis, K., Pitoura, E., Vassiliadis, P.: Adding context to preferences. In: Proc. ICDE, Istanbul, Turkey, pp. 846–855 (2007)
16. Torlone, R.: Two approaches to the integration of heterogeneous data warehouses. Distributed and Parallel Databases 23(1), 69–97 (2008)
17. Xin, D., Han, J.: P-cube: Answering preference queries in multi-dimensional space. In: Proc. ICDE, Cancún, México, pp. 1092–1100 (2008)