Brokering infrastructure for minimum cost data procurement based on quality - quantity models*

Alessandro Avenali[2], Paola Bertolazzi[1], Carlo Batini[3] and Paolo Missier[4]

1 - Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti" - CNR, Italy, bertola@iasi.cnr.it

2 - Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", Roma, Italy avenali@dis.uniroma1.it

3 - Dipartimento di Informatica, Sistemistica e Comunicazioni, Università di Milano Bicocca, Milano, Italy batini@bicocca.mi.it

4 - School of Computer Science, The University of Manchester, UK pmissier@cs.man.ac.uk

January 2007

Abstract

Inter-organization business processes involve the exchange of structured data across information systems. We assume that data are exchanged under given condition of quality (offered or required) and prices. Data offer may include bundling schemes, whereby different types of data are offered together with a single associated price and quality. We describe a brokering algorithm for obtaining data from peers, by minimizing the overall cost under quality

*The work presented in this paper has been partially supported by the eG4M MIUR FIRB project on e-Government in Mediterranean Countries and the MIUR FIRB MAIS project - Multi-channel Adaptive Information Systems: models, methodology, qualifying object-oriented platform and architectures for the flexible on-line information systems

requirements constraints. The algorithm extends query processing techniques over multiple database schemas to automatically derive an integer linear programming problem that returns an optimal matching of data providers to data consumers under realistic economic cost models.

Key words: Data quality, information market, information economics, quality cost optimization, bundle of data, brokering service, integer linear programming.

1 Introduction

For large businesses and public sector agencies, good management of information assets has long been a key to their effectiveness in delivering quality services to users, and many organizations have processes to manage the quality of their data. Recently, advances in the technology for large-scale deployment of information services, for example over service-oriented software infrastructures, have enabled cost-effective data exchange across organizations. In business terms, this means that it is becoming increasingly feasible for organizations to (i) purchase or otherwise acquire data from other peers, and (ii) exploit their own information assets for marketing purposes. These capabilities may be used to offer advanced services to users.

Thus, a general common trend is for organizations to acquire the information needed to support user services from third-parties. Several studies have analyzed the economic relevance of the potential information market. Public agencies have been found to be the greatest producers of information by far, and the information they create and disseminate is often relevant for both the private and public processes, products, and services. In [31] an analysis of the commercial exploitation of *public sector information* is presented both for the USA and the European Union (EU). The study shows that the economic value of the information market in the EU for year 2000 amounted

approximately to 10% of that of the US, where it was 750 billion dollars, and it recommended regulating the information market, to provide further incentives for the public sector information trading across and within member states.

To understand the implications of this trend, the size of the information market must be compounded with the issue of its *quality*, as a factor that will presumably affect the cost of data and hence the overall information market. Quality of data has been an issue since the nineties. General frameworks are available from the literature for describing data quality properties, or *dimensions* [35, 34]. For instance, *accuracy* characterizes how well data represents its corresponding real-world entities. Another main issue concerned with information market is represented by offering *bundles* of data, which are indivisible units of data, each one with a single associated price and quality level. In fact, both the cost structure behind the production and the selling of digital information goods, and the necessity of implementing anti-competitive strategies can induce more and more data providers to offer indivisible units of different types of data (for example [26]).

Focusing again on the public sector, it is well known that public agencies, in order to provide services to citizens and businesses, manage large registries with overlapping and heterogeneous data, and exchange large amounts of data flows.

Such a huge number of registries, from one side is characterized by a high overlap, from the other side they are usually managed and updated with different policies, resulting in different levels of accuracy and other quality dimensions. In many data intensive processes sources are combined, and it is important for agencies and private users to be able to choose and compose data on the basis of the desired target quality. In other terms, the availability of such overlapping sources of data may be seen as an opportunity for the data demand, that may use a *quality driven query processing* strategy [25] that builds the global data set on the basis of the differentiated offer of data

characterized by different qualities. Furthermore, the quality of data has a cost, and, at the same time, heavily influences the quality, the cost, and the revenues of the processes that use the data. While considering the relationship between the quality and cost of quality issues, some authors start their analysis from a parallel between the emerging information market and established markets for other goods [7], with the final purpose of defining criteria for data quality control and improvement. These activities, like for other types of goods, have a cost which is a component of the selling price. Furthermore, in order to conceive rational methodologies for improving the quality of data, several authors have proposed data quality cost classifications [12] and [23] and *cost/quality optimization procedures* [6] that investigate the various different types of cost of non quality of data. Issues of quality driven query processing and cost/quality optimization have been addressed only recently so far.

In this paper we propose a *brokering algorithm* that provides a *cost quality broker service* for facilitating the procurement of data from third parties, based on the assumptions that consumer interest for data is based both on its cost and on its quality, and that distinct data can be sold together in a bundle with a single associated quality and price. The algorithm, starting from: (i) the *offer of data* with possible bundling schemes from a set of providers, its quality and cost, (ii) the global, integrated knowledge on the information content offered by providers, and (iii) a query, that expresses the *data demand*, namely data requested by consumers and their quality, provides the optimal choice in terms of selected data, their quality and cost. We note that the broker service can be used as a decision support system for managers who have the responsibility of information acquisition activities.

The rest of the paper is organized as follows. In Section 2 the information procurement scenario underlying our approach is presented, together with a first overview of the algorithm is presented,

and basic definitions. The two phases of the algorithm, decomposition and optimization, are detailed in Sections 3 and 4, respectively. A discussion on related work is presented in Section 5; Section 6 concludes the paper.

2 Overview of the approach and basic definitions

2.1 Information Procurement Scenario and Cost Model

To describe the input/output behaviour of the cost-quality broker, we introduce an *information procurement scenario* and a *cost model*. We assume the following:

- the information market consists of a potentially large number of organizations; each organization may have a role both as data provider and consumer relative to other organizations;
- providers offer a description of the data they can procure, along with a cost model and the quality of the data offered;
- providers make their data available in *bundles*, as opposed to single data elements (see Section 1);
- consumers express their data demand as queries, along with constraints on the minimal acceptable quality level;
- for each type of data of interest to a consumer, more than one provider may be capable of fulfilling at least part of the demand.

A distinguishing feature of our information procurement scenario is the concept of bundle and the associated cost model. Several economic reasons motivate providers to offer bundles of

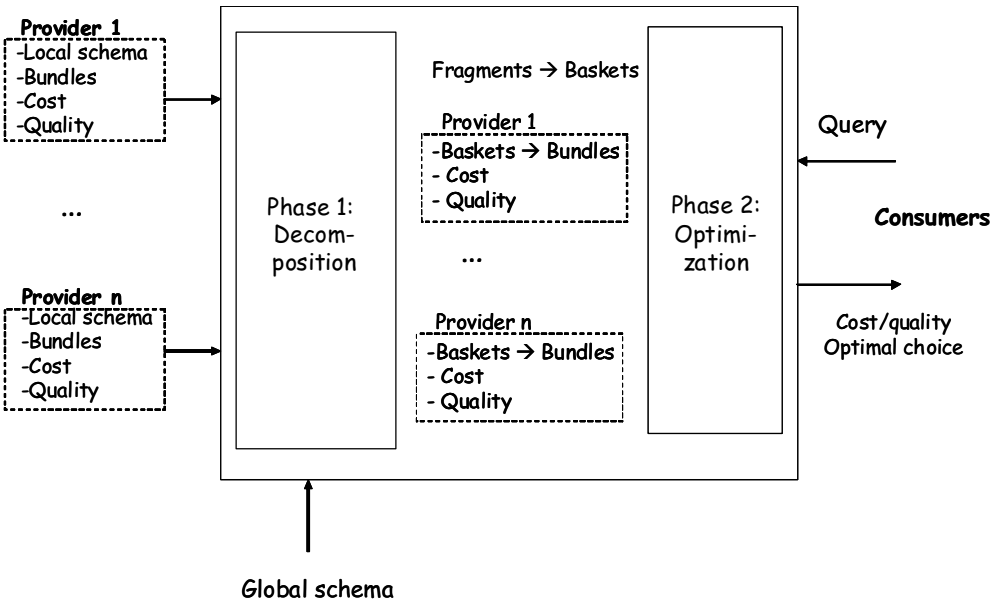
data. Digital information goods are typically characterized by high costs in the production and promotion of the first copy (high fixed cost of development), while additional copies are cheap to reproduce (low marginal cost). These goods are termed *non-rival*, because one's consumption does not limit the consumption of others. Since data have negligible marginal costs, bundling may be profitable [9, 13]. Bundles can also be used to provide discounts to consumers who acquire two or more (complementary) information goods. Moreover, bundling is a selling strategy that allows producers to reduce the uncertainty associated to the consumer's willingness to pay for individual goods [5, 17] (the willingness to pay for bundles exhibits lower variance than for separate individual items). Thus, the producer is able to extract more surplus from the consumers. Bundling results in a profitable pricing strategy even when the willingness to pay for goods is independent of the possible consumption of other goods in the bundle, and when there is no cost saving [1]. Finally, operators with significant market power may exploit bundling to engage in anti-competitive behavior [4, 26, 27].

2.2 Algorithm Overview

The two main steps in the brokering algorithm, namely *decomposition* and *optimisation*, are shown in Figure 1. We now provide a high level description of: (i) the data architecture for the brokering infrastructure, (ii) the inputs to the algorithm, (iii) the two phases, and (iv) the intermediate products exchanged by the two phases.

The underlying data architecture is grounded in the framework of federated database systems with a mediated query processing architecture [21]. We assume that data is described using the relational model, that each provider manages a local relational schema, and the local schemas are defined as mappings over a common *global schema*, that represents the whole available information

Figure 1: input/output two phase view of the brokering algorithm



content in an integrated view.

In this setting, the global schema is used by consumers to express their demand. The global query processor includes a *mediator*, an architectural component with functionality to recognize which of the local sources may contribute to the result of the global query and to translate the global query into a collection of local queries to be issued to the local sources. Each local query result represents a partial contribution to the global query result. The mediator formulates a query plan that includes the execution of the local queries, and then starting from these results, it produces a single consistent result to be delivered to the user.

The *decomposition phase* accepts and interprets a consumer query and related quality constraints. It decomposes each local query result into a set of *fragments*, in such a way that fragments from different local sources can be compared. This homogeneity allows classes of equivalent fragments, called *baskets*, to be defined across all participating providers. Fragments and baskets

will be formally defined in Section 2.4. The bundles are mapped to sets of fragments within each basket, along with their associated cost and quality.

The *optimization phase* formulates an *integer linear programming (ILP) model* to compute a cost-optimal combination of those partial results that is as complete as possible and satisfies all the constraints. The information on fragments, baskets, and bundles is translated into variables for an ILP model.

The results of this work extend and generalize those presented in [3]. In particular, our algorithm features the following innovative aspects:

- a query decomposition algorithm that is grounded in well-known research on distributed database systems [29] and federated database systems with a mediated query processing architecture [21];
- an algorithm that, starting from the partial results, automatically synthesizes the constraints and the objective function of a ILP;
- the new cost model considers a more general offer scenario.

We remark that the proposed brokering infrastructure is introduced as a mean to support customers in the process of data procurement in a complex offer environment, where different providers manage and offer a multitude of data sources. In particular, the brokering algorithm receives two essential inputs: the demand (a global query) and the offer scenario characterized by local schemas and exposed bundling conditions (with their price and quality features). However, in our work the offer scenario is assumed just as a given input of the brokering algorithm, in the sense that we do not care about how the suppliers decide (i) the offered prices, (ii) which bundles to sell, (iii) the quality discounts, etc. In fact, the strategic decisions of every data provider in terms of

prices, data bundling, etc. are the result of a complex stage, characterized by the maximization of the supplier' expected economic profit, which is not modelled in our work.

As far as the first and the second items are concerned, it is well known that the formulation of an optimization model is not an easy task, due to the lack of competence in most private and public organizations. In this paper we show that this task can be automated. Concerning the cost model, we extend the model proposed in [3], where only one price is associated with each offered bundle, by allowing suppliers to apply discounts in the case that multiple copies of the same bundle are demanded and/or lower quality levels for the data in the bundle are required.

2.3 Schema Mappings and Query Rewriting

In this section we introduce the concepts of global schema, local schema, and the type of mapping we define among them, according to the local as view (LAV) model. In our setting the *global schema* is a relational schema R in the relational model [11], defined over a set of attributes $\mathcal{A} = A_1, \ldots, A_n$. We will assume, without loss of generality, that the *primary key* of the global schema is the first attribute A_1.

In the running example, the global schema includes the following attributes:

`AllRes(`<u>`PersonId`</u>`, Province, Age, Income, Pathology, LastClassAttended)`

where the underlined attribute `PersonId` is the primary key.

The basic idea of the local as view (LAV) model [18, 21] is that each local schema is defined as a *view mapping*, that is, a relational expression on the global schema: $\pi_{\mathcal{A}_L}(\sigma_p(R))$, where symbol $\pi_{\mathcal{A}_L}$ denotes a *projection* relational operator defined on the set of attributes $\mathcal{A}_L$ of local schema L and symbol $\sigma_p(R))$ denotes a *selection* operator defined on the predicate p. The pair $[P, A_L]$ will be called in the following *selection projection condition* or, simply *condition*. Furthermore, we make

the assumption that, given pk, the primary key for R, for every pair of local schemas L_1 and L_2, the condition

$$pk \subseteq \mathcal{A}_{L_1} \cap \mathcal{A}_{L_2}$$

must hold. This means that, in order to merge data referring to the same tuples in the global schema, it is always possible to join tuples of different views through the same key.

This last assumption may seem in contrast with one of the main problems that arise when the quality of the data is questioned, i.e., that equivalent tuples in different relations may not be recognized as such, because their identifiers are slightly different. While the assumption that different schemas share the same primary key is indeed realistic, discrepancies in the key values in corresponding relations may make the join incomplete. Consider for example the two relations *Res-Mi* and *Adults* from Figure 2, and two tuples for the same individual, one in each relation. Unless *PersonId* has exactly the same value in the two tuples, they will not be joined.

Known as data de-duplication, or record linkage, this problem has long been studied by the data quality community, and a number of well-known de-duplication algorithms are now available –see for instance [8], chapter 5, and [10] for a recent survey on the topic. Furthermore, a number of techniques, based on similarity metrics, have been developed to perform *approximate joins* in the presence of discrepancies in the joined values [20]. Leveraging these results, we are going to assume that de-duplication has indeed been performed across schemas, by the providers or by some central agency, prior to the providers offering their data on the market.

To understand the idea underlying the LAV model, imagine that the global schema has been materialized, and that the following query is issued:

$$V = \pi_{Personid,Province,Age,Income,LastclassAttended}(\sigma_{Prov='MI'}(\text{AllRes}))$$

Figure 2: The four local schemas of the running example

Provider Id	Provider	Content	Local Schema	Mapping	
				Predicate p	Attribute set A_L
MP	Milan Province	Residents of Milan Province	Res-Mi(PersonId, Province, Age, Income, LastClass Attended)	Pro = 'MI'	12346
BD	Best Data	Adults (Age >= 18)	Adults(PersonId, Province, Age, Income, LastClass Attended)	Age >= 18	12346
LR	Lombardy Region	All residents in Lombardy	AllRes(PersonId, Province, Age, Income, Pathology)	true	12345
BP	Brescia Province	Residents of Brescia Province	Res-Bs(PersonId, Province, Age, LastClass Attended)	Pro = 'BS'	1236

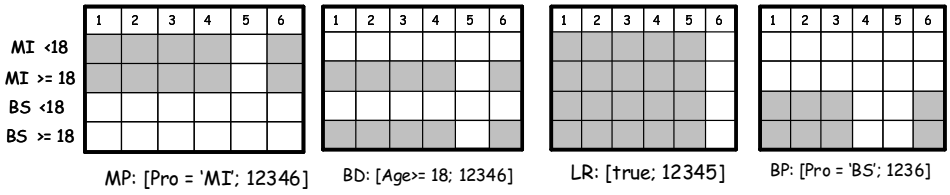
By saying that the query V is the LAV mapping from the global schema *AllRes* to a local schema *Milan province*, we say that the extension of *Milan province* can be obtained by computing V on the global schema *AllRes*. Of course, since in reality the materialized schema is that of *Milan province*, we may interpret the mapping as a specification of the contribution of a local schema to a possible materialization of the global schema.

The four local schemas correspond to three public providers, namely Milan province, the Lombardy region, and the Brescia province, and one private provider, Bestdata. The four providers are described in Figure 2 using the identifier, the name, the content description, the local schema, and the view mapping, expressed as a condition $[p, \mathcal{A}_L]$.

In Figure 3 we show the local schemas by means of a set oriented representation, where the part of the global schema present in the local schema is in gray; for example the local schema *MP* is defined for cells where the predicate Province = 'MI' holds, and only for attributes 12346 (on the left hand side of the figure the predicates corresponding to relevant groups of tuples appear with a shorthand notation).

According to LAV, given a set of mappings, the answer to a query Q requires a process of rewriting so that Q is expressed solely in terms of the mapping-defining views. The main theoretical

Figure 3: Set oriented representation of the four schemas



results indicate that (i) query processing is NP-complete in the worst case [22], and (ii) because the mappings can be incomplete, the goal of rewriting is to compute a maximal subset of the complete result, rather than a provably complete one (see for instance [18] for a thorough description of the problem). In brief, there are two main sources of complexity for this problem: (i) there are an exponential number of query rewritings, and (ii) testing containment for one such rewriting is itself NP-complete with respect to the length of the query. In our formulation, however, we make the simplifying but realistic assumption that none of the queries contain repeated predicates, making the problem linear. We further restrict queries to *conjunctive predicates* that are either (i) relational operators on ordered domains, of the form $x \ < relop > \ c$, of the types $=, \neq, <, \leq, >, \geq$ with c constant, or (ii) disjunctive predicates over set membership expressions, that is, $x =' c'$.

2.4 Fragments and Baskets

We investigate the issues related to query construction, and express the demand of data, defining the concepts of fragment and basket. Consider the following global query:

$$Q \equiv \pi_{PersonId,Age,Pathology}(AllRes) \tag{1}$$

and assume that the LAV mappings have been defined as in Figure 2. Several options exist for composing a global result from the local schemas. First, one can simply issue a single query to

provider LR alone, as follows:

$$\pi_{PersonId,Age,Pathology}(LR) \tag{2}$$

because LR provides all the required attributes and tuples.

A logically equivalent result can also be obtained by combining *fragments* of local schemas, that is, groups of values corresponding to to a given set of attributes and a set of tuples from some local schema. For example, the following fragments may be combined to obtain the complete result:

$$F_1 = \pi_{PersonId,Age}(BD)$$

$$F_2 = \pi_{PersonId,Pathology}(\sigma_{Age \geq 18}(LR))$$

$$F_3 = \pi_{PersonId,Age,Pathology}(\sigma_{Age < 18}(LR))$$

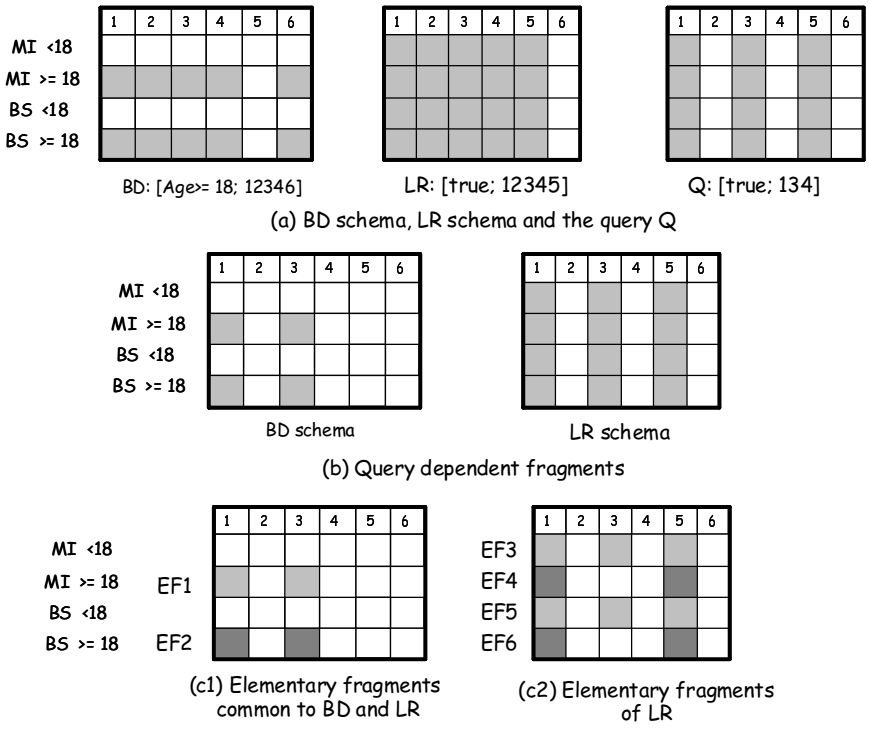
The result is expressed as:

$$(F_1 \bowtie_{PersonId} F_2) \cup F_3$$

For this composition to be feasible, we make use of the assumption made formerly, that local schemas include the common key `PersonID` as part of its attributes. Quality and cost are the criteria that drive the choice of fragments. In the example, we could prefer the query (2) if we know that provider LR owns good quality data for attributes `Age` and `Pathology`, and the cost of these data is not prohibitive. The second choice is preferred if we know that provider BD, for resident person with `Age` ≥ 18 bears ages of residents with better data qualities than LR and reasonable cost.

The space of all possible combinations of fragments can be visualized by overlapping the local schemas, based on their LAV mappings and the global query expression. We are looking here for *elementary fragments*, that is fragments that may make a contribution to the query and that it

Figure 4: Fragments resulting from comparison of BD and LR fragments in Step2



is not worthwhile to further decompose. In Figure 4 we see the elementary fragments that result from the comparison of: (i) the *LR* local schema, (ii) the *BD* local schema, and (iii) the query Q. Figure 4(a) shows the two schemas and the query Q; Figure 4(b) represents the fragments of the two schemas that independently may make a contribution to the query. Finally, Figure 4(c) shows the elementary fragments in two sets: on the left-hand side we see the fragments that are common to the two schemas, while on the right-hand side we see the fragments which belong to a single schema, namely the schema *LR*.

Notice that fragments EF_1 and EF_2 in Figure 4(c) are offered by both providers, while exactly one fragment from either of the two schemas should be used in a possible query to compose the result. Considering for example EF_1, we refer to the corresponding condition $[Pro = MI \wedge Age \geq 18; 13]$, generalized as $[p; A]$ as a set of (two) fragments called *basket*, whose

members are the two logically equivalent fragments:

$\pi_{PersonId,Age}(\sigma_{Pro=MI}(BD))$ and $\pi_{PersonId,Age}(\sigma_{Pro=MI \wedge Age \geq 18}(LR))$

Thus, the definition of a basket is intensional (a condition) and applies to several local schemas, while a fragment is a collection of (parts of) tuples obtained from the application of the basket expression to a specific schema.

2.5 Representing Quality and Bundles

We now extend our formalism to deal with data qualities and the demand and offer of data. Data quality deals with a wide number of different *dimensions*, that express properties of data. The most investigated dimensions are *accuracy* and *completeness*. Accuracy, as we have seen in Section 1 measures the closeness of the actual data values to their exact values. Completeness measures the extension of data in representing the real world. Measures or *metrics* can be assigned to dimensions: in relational tables, dimensions, and corresponding metrics, they can refer to tuples (e.g. a null value in a tuple results in low completeness) or else to the entire relation (e.g. only 80 % of the tuples are correct, resulting in an accuracy of 0.8), or else to attributes defined in the relation; for example if we know that values of the attribute `Age` are specified only in 90 % of cases, *null* otherwise, the value of completeness is 0.9. In this paper, dimensions are associated to attributes. Furthermore, we assume that for a relation with attributes $A_1, \ldots, A_m$, the data quality of the set of attributes is represented as a m-tuple q of vectors $q_i = (q_{i1}, \ldots, q_{in})$, one for each attribute A_i.

Two assumptions are made that allow us to compare quality-annotated relations across multiple providers. Firstly, as mentioned in Section 2.3, for each relation the primary key is considered correct and not *null*, so that relations from different providers can be joined successfully. This assumption, already made by other authors [28] in the context of data integration in the presence

of errors, is justified by the availability of de-duplication algorithms that can perform reconciliation prior to data marketing. Secondly, we assume that it is the responsibility of the providers to assess the quality vectors, and to be fair to consumers regarding their values. In making this hypothesis, we argue that enforcing providers' fairness with quality of data is in principle no different than with quality of service. This issue is normally addressed by assuming that trusted third parties have auditing authority over the providers, so that penalties can be levied when quality values are found to have been mis-reported.

We now have to define how data quality is expressed in the demand of data by consumers, and in the offer of data by providers. Consumers express quality constraints on the global schema, alongside the global query. Constraints are of the form $q_{ij} \ < relop > \ c_{ij}$, where q_{ij} refers to the value of quality dimension j for the attribute i and $< relop >$ is a relational operator. For example, considering query (1) above, the expression $q_{Age,completeness} > 0.6$ indicates the threshold value for dimension *completeness* relative to the `Age` global attribute.

From the provider's perspective, data sets and their quality are associated to bundles. A *bundle* is a triple $bu = (c, p, q)$ that specifies a relation the provider is committed to sell as: (i) a condition c on the local schema, expressed, as usual, as a pair $[p, A]$, (ii) its price c, and (iii) its quality vector q. A provider may declare several bundles, possibly overlapping in content. The following are valid bundles for provider *Lombardy region* (for conciseness, we denote here attributes with their identifiers):

i $bu_{LR1} = ([Prov =' MI' \vee Prov =' BS'; 12345], p_1, q_1)$,

ii $bu_{LR2} = ([Prov =' MI' \wedge Age < 18; 12345], p_2, q_2)$,

iii $bu_{LR3} = ([Prov =' MI' \wedge Age \geq 18; 12345], p_3, q_3)$,

iv $bu_{LR4} = ([Age \geq 18; 12345], p_4, q_4)$.

We assume that the quality of attributes in local schemas is homogeneous across the different parts of the local schema. Thus, by definition bundles inherit the quality vector associated to the local schema. Finally, quality also influences the composition of fragments in the process of query construction. When composing different relations with given quality dimensions and metrics with a union or join operation, we may wonder if functions exist that allow to automatically compute the values of metrics for the composed relation. Such composition functions have been investigated in the literature (see Section 5). Given two relations r_1 and r_2 with sizes respectively $|r_1|$ and $|r_2|$, in case of the union operator we assume as composition functions for quality dimensions *accuracy* and *completeness* (in short *qd*) the following expression:

$$qd(r_1 \bigcup r_2) = (|r_1| * qd(r_1) + |r_2| * qd(r_2))/(|r_1| + |r_2|)$$

The above formula provides an approximation of the correct value, since in general the two relations may overlap; see Section 5 for a more detailed discussion on comparison functions. In case of join, we simply have to juxtapose the quality vectors of the joined attributes.

3 Decomposition

In this section, we describe the first phase of the brokering algorithm in detail. The phase is composed of two steps. The goal of Step 1 is to find all fragments and corresponding baskets in local schemas that may potentially contribute to satisfy the data demand, expressed by a query Q. The goal of Step 2 is to relate the data demand, expressed by baskets, and the data offer, expressed by bundles procured by providers. For each basket we seek all the bundles that contain it, which can be used to satisfy the demand. In the following, the example global query is represented using

our shorthand notation:

$$Q \equiv \pi_{12346}(\sigma_{Age \geq 10}(AllRes)) \tag{3}$$

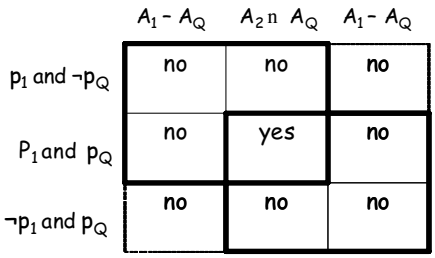
3.1 Finding Elementary Fragments and Baskets

In a traditional optimal plan algorithm for a distributed query [29], one is interested in minimizing the cost of transmission. Thus, given a selection projection query Q whose extension is the fragment F, the goal is to find all conditions expressing fragments that *contain* F. Here we have to solve a different task, namely, to find all fragments that may contribute to the final result and consequently *are contained in* F. They can also be optimally *composed*, knowing their cost and quality. Indeed, only within this setting we are free to choose the most suitable mix of fragments to fit the quality/cost problem.

The step consists of two sub-steps. The goal of Step 1.1 is to find the set of fragments in the local schema for each local schema that may contribute to the final query. When comparing the query condition and the local schema condition of a local schema V_1, three cases may arise:

- Case 1: the local schema fragment is contained in the query fragment; in this case the local fragment is selected;
- Case 2: the query fragment is contained in the local schema fragment; here the query fragment is selected;
- Case 3: there is a non null intersection between the two fragments. This case is expressed more formally in Figure 5, where compared fragments are marked with bold lines and all possible combinations of predicates p_Q and p_1 and set relationships among sets of attributes A_1 and A_Q are shown. In this case only the sub fragment indicated in Figure 5 with *yes* has

Figure 5: The case of partial overlapping between query and local schema conditions



to be selected, since this is the unique contribution of the local schema to the query.

In Figure 6 we see the fragment corresponding to the query and the four fragments resulting from application of Step 1.1 to the four local schemas. The goal of Step 1.2 is to relate all fragments obtained after Step 1.1 in order to find the *elementary fragments*, namely fragments that (i) contribute to the query, (ii) it would not be useful decompose further, and (iii) do not contain other fragments. Also in this step three cases are given. Case 1 and Case 2, corresponding to one fragment containing the other one, do not result in new fragments. Case 3, corresponding to a partial overlapping of fragments, has the possible subcases as shown in Figure 7. In this case we have to include all fragments denoted with *yes* in the figure, since they are all part of the two original fragments, and may contribute to the query result. The two cases denoted as *no* are to be excluded, since they do not belong to either of the original fragments. In Figure 8 we see the elementary fragments resulting from the comparison of *LR* tp *BD* local schemas, subdivided in common and uncommon fragments.

In Step 1.3 we collect all elementary fragments having the same condition expression $[p; A]$ and consequently, correspond to the same basket. In Figure 9 we show baskets and corresponding providers for the fragments considered in Figure 8. The algorithm requires to test predicate containment, an NP-complete operation in the most general case, which we simplify to linear operations

Figure 6: Fragments resulting from Step 1.1

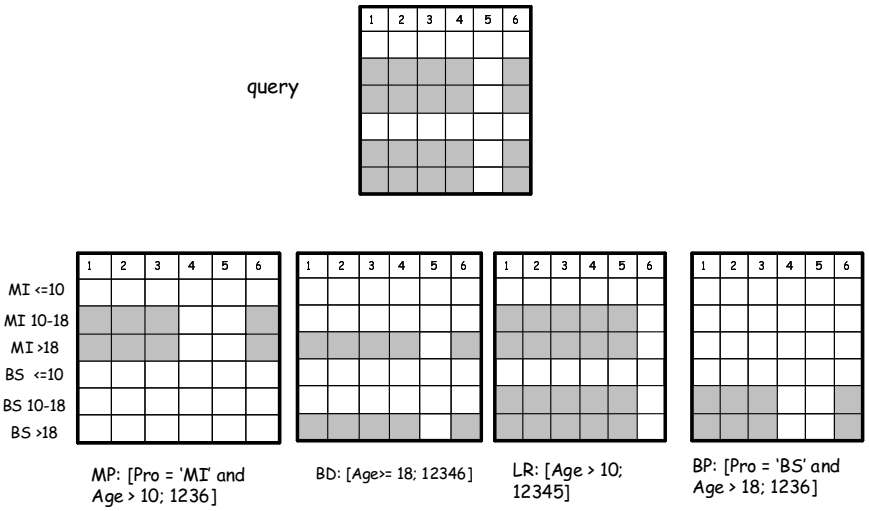


Figure 7: The case of partial overlapping between conditions of demand fragments

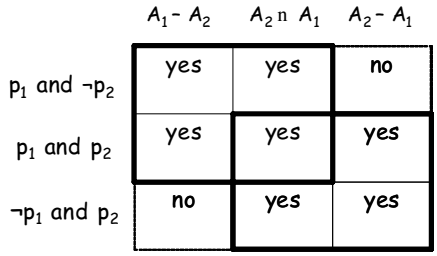


Figure 8: Fragments resulting from comparison of BD and LR fragments in Step 2

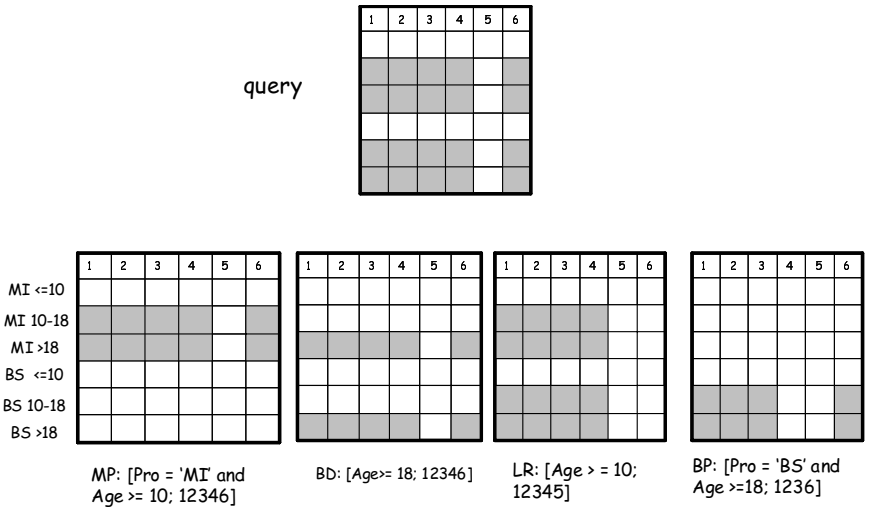


Figure 9: Baskets and corresponding providers for the example of Figure 8

Basket	Basket condition	Providers
bk_1	Pro = 'MI' and Age >= 18; 1234	BD, LR
bk_2	'Pro = 'BS' and Age >= 18; 1234	BD, LR
bk_3	Pro = 'MI' and 10 < = Age < 18; 1234	LR
bk_4	Pro = 'MI' and Age >= 18; 16	BD
bk_5	Pro = 'BS' and 10 < = Age < 18; 1234	LR
bk_6	Pro = 'BS' and Age >= 18; 16	BD

by considering only the type of logical operators already mentioned.

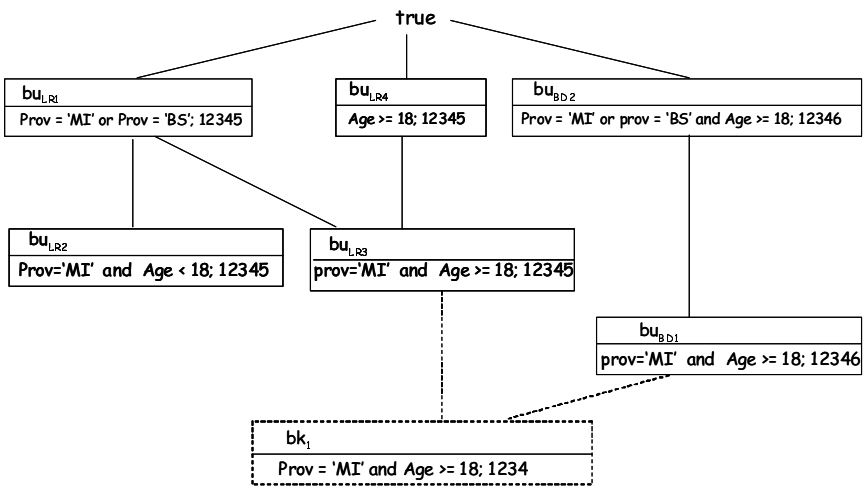
3.2 Relating Elementary Baskets to Bundles

We recall from Section 2.1 that a provider offers its local data in bundles of the form $bu = (c, p, q)$, which specify a data set as a local condition c, its price p and its quality vector q. To understand their relationship to the basket formulation described so far, note that bundles are atomic units of data that can be purchased, and thus they represent a "business view" of a provider's offering, which is independent of any query. On the other hand, the query processor provides a description of the local data in terms of atomic baskets, determined by the query, from which individual fragments can be selected. Selecting one fragment from each basket provides a coverage of the query result.

These two views are reconciled by mapping a basket onto one or more bundles, so that selecting a basket results in a choice of bundles that can be purchased. Here we make the assumption that for each basket, at least one bundle exists that contains it. This assumption looks reasonable, since providers tend to offer large data sets.

Carrying on with our example, assume that provider *Lombardy region* sells the four bundles

Figure 10: Baskets and their relation with bundles



described in Section 2.5 and provider *Bestdata* defines two bundles:

v $bu_{BD1} = ([Prov =' MI' \wedge Age \geq 18; 12346], p_5, q_5)$

vi $bu_{BD2} = ([Prov =' MI' \vee Prov =' BS' \wedge Age \geq 18; 12346], p_6, q_6)$

Consider basket $bk_1 = [Prov =' MI' \wedge Age \geq 18; 1234]$ owned by providers *Bestdata* and *Lombardy region*. We may represent the many-to-mant containment relationship between baskets and bundles using a lattice, as shown in Figure 10, where the dotted box is a basket. The algorithm has the following choices for bundles: (i) bu_{LR1} from provider *Lombardy region*, (ii) bu_{LR3} from provider *Lombardy region*, (iii) bu_{BD1} from provider *Bestdata*, (iv) bu_{BD2} from provider *Bestdata*. Note that bundles bu_{LR1} and bu_{BD2} lead to additional, unrequested data to be purchased with respect to bundles bu_{LR3} and bu_{BD1}, but they may still be convenient in terms of quality and price. The query processor may then filter out the unrequested data, or it may decide to retain it and use it in combination with other bundles.

4 Optimization

The definition of a bundle can be generalized slightly at this point, by adding the notion of discounts for bundles of data both in terms of provided quality and number of sold copies. Thus, a bundle is now a quadruple $bu = (c, p, q, qt)$, where qt copies of the data represented by the condition c, characterized by a quality vector q, are offered at price p. However in the running example, for simplicity of notation, we will keep on considering only prices related to quality.

The optimization phase consists of first formulating and then solving the problem of selecting fragments across baskets in such a way that quality and quantity requirements are satisfied, and the total cost to acquire bundles containing such fragments is minimized. Firstly a graph is constructed, then an ILP formulation is derived from this graph, and finally an optimization solver is applied to find an optimal solution.

Selecting a bundle bu means that, for every basket bk in the *containing bundles* relationship with bu, the fragment supplied by the same provider of bu in bk must be chosen. We say that these fragments are *associated* to the bundle bu. In our framework, when a consumer submits a global query he/she also specifies, for every demanded attribute A_k ($k = 1, \ldots, l$), the quality q_k and quantity qt_k required for A_k. Note that, without loss of generality, we assume that the consumer demands the first l attributes of the global schema. We denote by D_k the set of baskets whose fragments cover the demanded attribute A_k (i.e. the demanded attribute A_k is provided by selecting exactly one fragment from each basket in D_k). A complete description of the relevant offer for the submitted global query with quality and quantity requirements has to take into account information deriving from local schemas and exposed bundling conditions. We denote the offer scenario by a pair $(\mathcal{B}K, \mathcal{B}U)$ where (i) $\mathcal{B}K = \{bk_1, bk_2, \ldots\}$ is the set of the generated baskets, and

(ii) $\mathcal{BU} = \{bu_1, bu_2, \dots\}$ is the set of the offered bundles.

For instance, if we consider the running example, for the data providers *Lombardy region* and *Bestdata* the first phase of the algorithm returns:

- the set of baskets $\mathcal{BK} = \{bk_1, bk_2, bk_3, bk_4, bk_5, bk_6\}$, defined in Section 3.2;
- the set of bundles $\mathcal{BU} = \{bu_{LR1}, bu_{LR2}, bu_{LR3}, bu_{LR4}, bu_{BD1}, bu_{BD2}\}$, defined in Sections 2.5 and 3.2.

Given a scenario demand characterized by A_k, q_k, qt_k for $k = 1, \dots, l$ and the sets $(\mathcal{BK}, \mathcal{BU})$, we want to find a set of bundles satisfying all quality and quantity requirements with the minimum cost. We define this problem as the *minimum cost supplying* problem (MCS for short). We now provide a formulation of MCS. In order to represent the offer and demand scenarios we use a tripartite graph $G = (W, U, V, E)$, where:

- W is the set of vertices representing the *bundles* with the associated price matrices,
- U is the set of vertices representing the *fragments*,
- V is the set of vertices representing the *baskets*,
- $E = E_1 \cup E_2$ is the edge set such that E_1 (E_2) is the set of edges connecting W (V) with U.

The graph is constructed as follows. We associate exactly one vertex in W to every bundle $bu \in \mathcal{BU}$. Furthermore, given the vertex $w_i \in W$, we denote by qt_i, q_i and pr_i, respectively, the related quantity, the quality vector and the price characterizing the bundle bu associated to w_i. For every $w_i \in W$, each fragment associated to the bundle corresponding to w_i is represented by exactly one vertex $u_j \in U$. We recall from Section 2.5 that q_i is a m-tuple of quality vectors, each one associated to a specific attribute of the bundle. Given the fragment corresponding to u_j, we may associate

to u_j an m-tuple of quality vectors denoted as $\tilde{q}_j$, obtained projecting q_i on the attributes defined in u_j. In particular, in $\tilde{q}_j$ every demanded attribute A_k of the fragment u_j is characterized by a quality vector $\tilde{q}_{j,k}$ (whose size depends on the number of quality dimensions). However, from now on, for the sake of simplicity and w.l.o.g., we limit the number of quality dimensions to one. Hence, the scalar $\tilde{q}_{j,k}$ is associated to every demanded attribute A_k in the fragment u_j. To keep a trace of the relation between the bundles and the associated fragments, an edge (i, j) between vertices w_i and u_j is added in E_1. Note that, on the supplier side, there is one vertex u for each fragment of every vertex w (this implies that multiple copies of a fragment can be represented in the graph). For each basket $bk_h \in \mathcal{B}K$ a vertex $v_h \in V$ is introduced. An edge (h, j) between vertices v_h and u_j belongs to E_2 iff the fragment represented by u_j is a member of the basket bk_h.

Coming back to the running example, the resulting tripartite graph $G = (W, U, V, E)$ is shown in Figure 11. Observe that, in order to improve the readability of the figure and w.l.o.g., we restrict our running example to two attributes, say A_4 and A_6.

Given the graph $G = (W, U, V, E)$, we provide an ILP formulation of the MCS problem. A binary variable x_i is associated to each vertex $w_i \in W$ and it assumes 1 iff the corresponding bundle is selected to be sold. Moreover, a binary variable y_j is associated to each vertex $u_j \in U$ and it is equal to 1 iff the corresponding fragment is chosen to satisfy the consumer demand. We also denote by r_j the size of the fragment u_j. The problem can thus be formulated as follows:

$$\min \sum_{i:w_i \in W} pr_i \cdot x_i$$

subject to the constraints:

$$y_j \leq x_i \qquad (i, j) \in E_1 \tag{4}$$

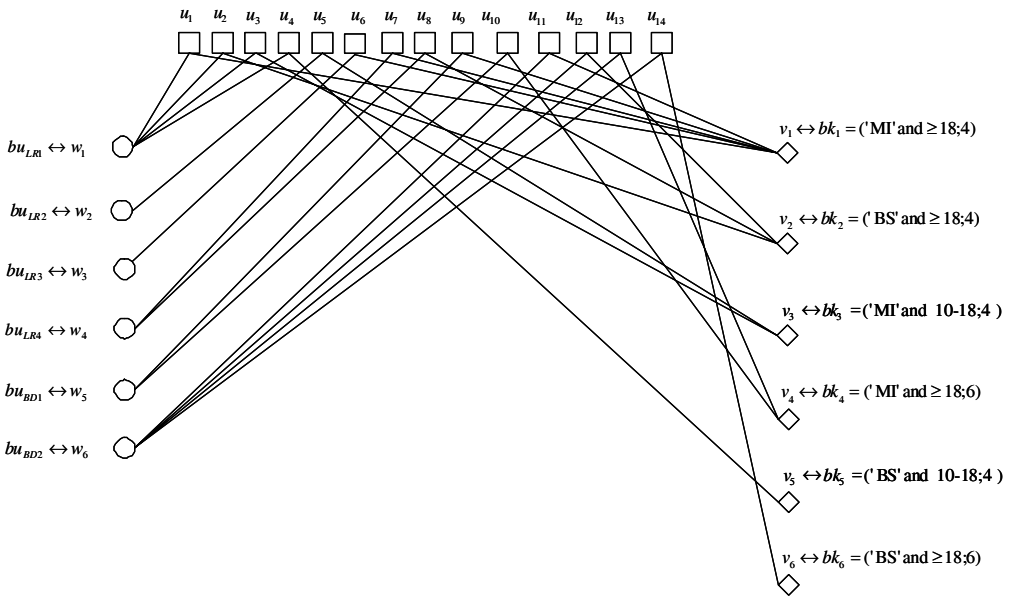


Figure 11: Tripartite graph

$$\sum_{j:(h,j)\in E_2, h:bk_h\in D_k} y_j = qt_k \qquad k \in \{1,\ldots,l\} \tag{5}$$

$$\sum_{j:(h,j)\in E_2, h:bk_h\in D_k} \tilde{q}_{j,k} \cdot r_j \cdot y_j \geq q_k \cdot \sum_{j:(h,j)\in E_2, h:bk_h\in D_k} r_j \cdot y_j \qquad k \in \{1,\ldots,l\} \tag{6}$$

$$x_i, y_j \in \{0,1\} \qquad i : w_i \in W, j : u_j \in U \tag{7}$$

Constraint (4) means that if a fragment from a bundle is chosen, then the entire bundle is chosen. Constraint (5) says that exactly qt_k fragments containing demanded attribute A_k must be chosen among baskets in order to satisfy the consumer's quantity requirements. Constraint (6) means that if a demanded attribute can be obtained as a merge of fragments (each in a different basket), then the overall quality of the merge is given by the weighted average of the elementary fragment qualities. Moreover, constraint (6) allow us to apply the composition functions discussed for accuracy and

completeness in section 2.5. Obviously, in the case that quality dimension is larger than one, constraint (6) must be replicated for each quality dimension.

The provided formulation takes inspiration from the mathematical model underlying the facility location problem ([15]) with no connection costs, where a given demand (computed baskets) has to be served by facilities (data fragments) which can be installed in several locations (the data bundles offered, whose associated prices represent facilities costs); this problem, as known, is NP-hard. Note, however, that data differ from physical commodities in that the "overlapping" of their sources represents an opportunity to create a marketplace by diversifying the offer (e.g. by cost and quality), rather than a potential for wasting resources.

5 Related Work

The impact of information quality on actual decision quality using a theoretical and a simulation model is investigated in [32]. Relevant work has been done toward associating quality properties to relational databases, and using that information during query processing. A distinction should be made between two issues (i) *Quality composition* defines an algebra for composing data quality dimension values in queries,

and (ii) *Quality-driven query answering*, the task of providing query results on the basis of a quality characterization of data at sources, and possibly cost of data.

As discussed in section 2.5, in this paper we adopt approximate formulae for composition functions, referring to the union and join operators. Quality composition has been addresses in several papers, Wang et al. [35], Naumann et al. [28], Scannapieco et al. [33], and Parssian et al. [30]. Several characterizations exist for two dimensions, namely accuracy and completeness.

Concerning accuracy, [35] distinguishes between a *relation accuracy* and a *tuple accuracy*. Several results are provided for selection and projection operators. Results provided in [30] are richer, and concern compositions in terms of cartesian product, projection, selection, and join operations.

Referring to completeness, major contributions are [28], and [33]. In the first paper the authors are interested to evaluate the quality of the process of composing sources, in order to put together data that is split in different relational tables. Formulae for completeness composition functions are provided for several possible set containment relationships among sources. Similar results, extended to the union and difference operators are described in [33], within a different model, in which two different hypothesis are made on the (i) coincidence, or (ii) difference in the domain to which the two sources refer.

Regarding the second topic, that is, quality driven query answering, in the context of mediator-based data integration, Florescu, Koller, and Levy (1997) [14] deal with the completeness problem by introducing various probability distributions regarding the content overlap across multiple database sources, and efficient ways to compute them.

With respect to our research, this work appears to only include a quality scoring model, but not a cost model, hence the selection of the best plan is essentially based on a quality ranking of the candidate plans.

The first setting of the quality cost optimization problem is due to Avenali, Batini, Bertolazzi, and Missier [3]. Successively the problem, to the best of our knowledge, has been addressed only in [2]. In this approach, in order to obtain the required data, customers must buy multiple data sets from different providers and then clean and merge them. In this case a broker architecture intermediates between users and syndicated data providers.

There are several differences between the approach presented in the above cited paper and the

one presented in this paper. First, in our paper, in the decomposition phase, all candidate fragments are built, while in [2] fragments are given a priori, and the broker has no other choice than to manipulate them. Second, we have a concept of bundle, as a further constraint in the optimization process. Third, authors in [2] provide a non linear formulation of the problem of identifying a family of data sets to satisfy the query with a minimum overall price. However, the solutions found are not guaranteed to be optimal.

For the optimization phase of the broker service two main areas must be cited. The first concerns the strategy of bundles, that has been discussed in the introduction. The second concerns the algorithms to solve ILPs similar to the one proposed in our paper.

The facility location problem underlying in the structure of the ILP presented in Section 4 (to minimize the cost of data procurement in a given offer scenario) is a widely studied topic in the operations research literature [24]. There are a number of papers that concern exact methods for this problem [19, 16]. Most of exact methods can be straightforwardly applied to our case.

6 Conclusions and extensions

We have presented a brokering algorithm that supports managers in the process of buying information from multiple data sources, that are characterized by different cost and quality. The algorithm accepts (i) a collection of quality vectors, one for each record in the sources, and (ii) a query over a global schema, as well as the mappings from local to global schema (in a local-as-view setting). It computes the most complete answer to the global query with the best cost-quality ratio.

The algorithm consists of two phases. During the first phase, using the schema mappings, a set of local fragments for the query result are identified. In the second phase a variable is associated to

each fragment, while their corresponding quality and cost are used to formulate constraints for an ILP problem. The problem solution contains a complete answer obtained under quality constraints, and at minimal cost. The first phase includes a particular case of a query subsumption problem. However the simplifying assumptions on the conditions make it polynomial. Although the second phase is known to be NP-complete, the size of the problem which is determined by the number of providers and of local schemas is expected to be small. The algorithm in practice is meant to be used by decision-makers with the responsibility of acquiring quality data from third parties.

The query answer is computed under the assumption that the quality vectors, supplied by the data provider reflect the best quality information known to the provider. In reality, providers may provide misleading quality information to their advantage, and consumers have normally no way to verify this information during the course of query execution. As mentioned in Section 2.5, this issue is normally addressed by assuming that trusted third parties have auditing authority over the providers, and that they can issue penalties when the information is found to be untruthful.

The algorithm can be usefully extended to support a *coordinated spot market*, where multiple consumers simultaneously require portions of data with specified quality levels, and multiple suppliers submit their offers and associated quantity-quality matrices to a Central Public Supplier (CPS) mediator. For instance, the CPS might be in charge of selling data owned by multiple local public agencies to individuals, businesses and other public agencies. In this case, in order to exploit the quantity/quality discounts as much as possible, the CPS could coordinate the purchasing process by collecting and then matching the overall demand and offer. In particular, the problem of allocating offered data among consumers can be formulated as a simple extension of the ILP presented in Section 4. We are interested in implementing the DSS presented in this paper and to develop the whole model underlying the coordinated spot market outlined above.

References

[1] W.J. Adams, J.L. Yellen, Commodity Bundling and the Burden of Monopoly, Quarterly Journal of Economics, 1976, vol. 90 (3).

[2] D. Ardagna, C. Cappiello, M. Comuzzi, C. Francalanci, B. Pernici, A Broker for Selecting and Provisioning High Quality Syndicated Data - International Conference on Information Quality, MIT, Boston, 2005.

[3] A. Avenali, P. Bertolazzi, C. Batini, P. Missier, A Formulation of the Data Quality Optimization Problem in Cooperative Information Systems, in: Proceedings of the International Workshop on Data and Information Quality (DIQ 2004) in conjunction with CAiSE 04, Riga, Latvia, 2004.

[4] I. Ayres, B. Nalebuff, Going Soft on Microsoft? The EU's Antitrust Case and Remedy, The Economists' Voice, 2005, vol. 2 (2).

[5] Y. Bakos, E. Brynjolfsson, Bundling Information Goods: Pricing, Profits and Efficiency, Management Science, 1999, 45 (12), pp. 1613–1630.

[6] D. Ballou, H. Pazer, Designing Information Systems to Optimize the Accuracy-Timeliness Tradeoff, Information Systems Research, vol. 6 (1).

[7] D.Ballou, R.Wang, H.Pazer, G.K.Tayi, Modelling Information Manufacturing Systems to Determine Information Product Quality, Journal of Management Sciences, vol. 44 (4).

[8] C. Batini, M. Scannapieco, Data Quality: Concepts, Methodologies and Techniques, Springer, Data-Centric Systems and Applications, 2006.

[9] C.H. Brooks, R. Das, J.O. Kephart, J.K. MacKie-Mason, R.S. Gazzale, E.H. Durfee, Information Bundling in a Dynamic Environment, in: Proceedings of the IJCAI-01 Workshop on Economic Agents, Models, and Mechanisms, 2001, Seattle, Washington.

[10] A. K. Elmagarmid, P. G. Ipeirotis, V. S. Verykios, Duplicate Record Detection: A Survey, IEEE Transactions on Knowledge and Data Engineering, vol. 19, No 1, January 2007.

[11] R. Elmasri and S. Navathe, Fundamentals of Database Systems, Benjamin and Cummings, 2002.

[12] L. P. English, Improving Data Warehouse and Business Information Quality: Methods for Reducing Costs and Increasing Profits, 1st Edition, John Wiley & Sons, 1999.

[13] S.A. Fay, J.K. MacKie-Mason, Competition Between Firms that Bundle Information Goods, in: 27th Annual Telecommunications Policy Research Conference, 2001, Alexandria, VA.

[14] D. Florescu, D. Koller, A. Y. Levy, Using Probabilistic Information in Data Integration, in: M. Jarke, M. J. Carey, K. R. Dittrich, F. H. Lochovsky, P. Loucopoulos, M. A. Jeusfeld (Eds.), Proceedings of 23rd International Conference on Very Large Data Bases (VLDB), August 25-29, 1997, Athens, Greece, Morgan Kaufmann, 1997, pp. 216–225.

[15] R.L. Francis, L.F. McGinnis, J.A. White, Locational Analysis, European Journal of Operational Research, 1983, vol. 12, pp. 220–252.

[16] R. Galvao, The Use of Lagrangean Relaxation in the Solution of Uncapacitated Facility Location Problems, Location Science, 1993, vol. 1, pp. 57–79.

[17] X. Geng, M.B. Stinchcombe, A.B. Whinston, Bundling Information Goods of Decreasing Value, Management Science, 2005, vol. 51 (4), pp. 662–667.

[18] A. Y. Halevy, Answering Queries Using Views: a survey, VLDB Journal, vol. 10 (4) (2001), pp. 270–294.

[19] K. Holmberg, M. Ronnqvist, D. Yuan, An Exact Algorithm for the Capacitated Facility Location Problem with Single Sourcing, European Journal of Operational Research, 1999, vol. 113, pp. 544–559.

[20] N. Koudas, D. Srivastava, Approximate Joins: Concepts and Techniques, Very Large Data Bases Conference, 2005, p. 1363.

[21] M.Lenzerini, Data integration: A Theoretical Perspective, in: Principles Of Database Systems, 2002, pp. 233–246.

[22] A. Y. Levy, A. O. Mendelzon, Y. Sagiv, D. Srivastava, Answering Queries Using Views, in: Principles Of Database Systems (PODS), 1995, pp. 95–104.

[23] D. Loshin Enterprise Knowledge Management - The Data Quality Approach - Chapter 4, Knowledge intelligence incorporated, 2004.

[24] P. Mirchandani, R. Francis, Discrete Location Theory, John Wiley and Sons, New York, 1990.

[25] A.Motro, P.Anokhin, A. Acar, Utility-based Resolution of Data Inconsistencies, in: F. Naumann, M. Scannapieco (Eds.), International Workshop on Information Quality in Information Systems 2004 (IQIS'04), ACM, Paris, France, 2004.

[26] B. Nalebuff, Competing Against Bundles, in: P. Hammond, G. Myles (Eds.), Incentives, Organization, and Public Economics, 2000, Oxford University Press.

[27] B. Nalebuff, Bundling as an Entry Barrier, Quarterly Journal of Economics, 2004, vol. 119 (1).

[28] F. Naumann, J.C. Freytag and U. Leser, Completeness of Integrated Information Sources, Information Systems, 29(7), 2004, pp.583-615.

[29] T. Oszu, P. Valduriez, Principles of Distributed Database Systems (Second ed.) Prentice Hall, Englewood Cliffs, N.J., 1999.

[30] A. Parssian, S. Sarkar, and V.S. Jacob, Assessing Data Quality for Information Products: Impact of Selection, Projection, and Cartesian Product, Management Science, vol. 50, No. 7, July 2004.

[31] Pira International, Commercial Exploitation of Europe's Public Sector Information, Final report for the European Commission, Directorate General for the Information Society, October 2000.

[32] S. Raghunathan, Impact of Information Quality and Decision-maker Quality on Decision Quality: a Theoretical Model and Simulation Analysis, Decision Support Systems, volume 26, Issue 4, October, 1999, pp. 275-286.

[33] M. Scannapieco, C. Batini, Completeness in the Relational Model: a Comprehensive Approach - Proceedings of the International Conference on Data Quality, Boston Ma November 2004.

[34] R. Wang, D. Strong, Beyond Accuracy: what Data Quality Means to Data Consumers, Journal of Management Information Systems 12 (4).

[35] R.Y. Wang, M. Ziad, and Y.W. Lee, Data Quality, Kluwer Academic Publisher, 2001, pp. 63-77.