On the Expressive Power of Data Integration Systems

Andrea Calì, Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini

Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza", Via Salaria 113, I-00198 Roma, Italy
{cali,calvanese,degiacomo,lenzerini}@dis.uniroma1.it
http://www.dis.uniroma1.it/~[cali,calvanese,degiacomo,lenzerini]

Abstract. There are basically two approaches for designing a data integration system. In the global-as-view (GAV) approach, one maps the concepts in the global schema to views over the sources, whereas in the local-as-view (LAV) approach, one maps the sources into views over the global schema. The goal of this paper is to relate the two approaches with respect to their expressive power. The analysis is carried out in a relational database setting, where both the queries on the global schema, and the views in the mapping are conjunctive queries. We introduce the notion of query-preserving transformation, and query-reducibility between data integration systems, and we show that, when no integrity constraints are allowed in global schema, the LAV and the GAV approaches are incomparable. We then consider the addition of integrity constraints in the global schema, and present techniques for query-preserving transformations in both directions. Finally, we show that our results imply that we can always transform any system following the GLAV approach (a generalization of both LAV and GAV) into a query-preserving GAV system.

1 Introduction

Data integration is the problem of combining the data residing at different sources, and providing the user with a unified view of these data, called global (or, mediated) schema [9]. The global schema is therefore the interface by which users issue their queries to the system. The system answers the queries by accessing the appropriate sources, thus freeing the user from the knowledge on where data are, and how data are structured at the sources.

The interest in this kind of systems has been continuously growing in the last years. Many organizations face the problem of integrating data residing in several sources. Companies that build a Data Warehouse, a Data Mining, or an Enterprise Resource Planning system must address this problem. Also, integrating data in the World Wide Web is the subject of several investigations and projects nowadays. Finally, applications requiring accessing or re-engineering legacy systems must deal with the problem of integrating data stored in pre-existing sources.

S. Spaccapietra, S.T. March, and Y. Kambayashi (Eds.): ER 2002, LNCS 2503, pp. 338–350, 2002.

The design of a data integration system is a very complex task, which comprises several different issues [10]. One of the most important aspect is the specification of the mapping between the global schema and the sources, and the use of such a specification for carrying out query processing.

Two basic approaches have been used to specify the mapping between the sources and the global schema [9, 11, 12]. The first approach, called *global-as-view* (or simply GAV), requires that the global schema is expressed in terms of the data sources. More precisely, to every element of the global schema, a view over the data sources is associated, so that its meaning is specified in terms of the data residing at the sources. The second approach, called *local-as-view* (LAV), requires the global schema to be specified independently from the sources. In turn, the sources are defined as views over the global schema. The relationships between the global schema and the sources are thus established by specifying the information content of every source in terms of a view over the global schema.

Intuitively, the GAV approach provides a method for specifying the data integration system with a more procedural flavor with respect to the LAV approach. Indeed, whereas in LAV the designer of the data integration system may concentrate on specifying the content of the source in terms of the global schema, in the GAV approach, one is forced to specify how to get the data of the global schema by queries over the sources.

A comparison of the LAV and the GAV approaches is reported in [16]. It is known that the former approach ensures an easier extensibility of the integration system, and provides a more appropriate setting for its maintenance. For example, adding a new source to the system requires only to provide the definition of the source, and does not necessarily involve changes in the global view. On the contrary, in the GAV approach, adding a new source may in principle require changing the definition of the concepts in the global schema.

It is also well known that processing queries in the LAV approach is a difficult task [15, 16, 8, 1, 7, 3, 4]. Indeed, in this approach, the only knowledge we have about the data in the global schema is through the views representing the sources, and such views provide only partial information about the data. Since the mapping associates to each source a view over the global schema, it is not immediate to infer how to use the sources in order to answer queries expressed over the global schema. Thus, extracting information from the data integration system is similar to query answering with incomplete information, which is a complex task [17]. On the other hand, query processing looks much easier in the GAV approach, where we can take advantage that the mapping directly specifies which source queries corresponds to the elements of the global schema. Indeed, in most GAV systems, query answering is based on a simple unfolding strategy.

Besides the above intuitive considerations, a deep analysis of the differences/similarities of the two approaches is still missing. The goal of this paper is to investigate on the relative expressive power of the LAV and the GAV approaches. In particular, we address the problem of checking whether a LAV system can be transformed into a GAV one, and vice-versa. Obviously, we are

interested in transformations that are equivalent with respect to query answering, in the sense that we want that every query posed to the original system has the same answers when posed to the new system. To this end, we introduce the notion of query-preserving transformation, and the notion of query-reducibility between classes of data integration systems. Results on query reducibility from LAV to GAV systems may be useful, for example, to derive a procedural specification from a declarative one. Conversely, results on query reducibility from GAV to LAV may be useful to derive a declarative characterization of the content of the sources starting from a procedural specification.

We study the problem in a setting where the global schema is expressed in the relational model, and the queries used in the integration systems (both the queries on the global schema, and the queries in the mapping) are expressed in the language of conjunctive queries. We show that in such a setting none of the two transformations is possible. On the contrary, we show that the presence of integrity constraints in the global schema allows reducibility in both directions. In particular, inclusion dependencies and a simple form of equality-generating dependencies suffice for a query-preserving transformation from a LAV system into a GAV one, whereas single head full dependencies are sufficient for the other direction. Finally, we introduce the GLAV approach, where both LAV and GAV assertions are allowed in the mapping, and illustrate how to adapt the technique from LAV to GAV to devise a query-preserving transformation from GLAV to GAV.

Also, the results presented in the paper shows that techniques for answering queries under integrity constraints are relevant in data integration. In particular, several approaches to answering queries under different forms of dependencies have been proposed in the last years (see for example [14]). Our results imply that these approaches can be directly applied to query answering in LAV, GAV, and GLAV systems with inclusion dependencies. Data integration is thus a good candidate as an application for experimenting these techniques in real world settings.

The paper is organized as follows. In Section 2 we describe the formal framework we use for data integration, and we introduce the notions of query-preserving transformation, and of query-reducibility between classes of data integration systems. In Section 3 we show that in the relational model without integrity constraints, the classes of LAV and GAV systems are not mutually query-reducible. In Section 4 we present the results on query-reducibility in the case where integrity constraints are allowed in the global schema. Finally, Section 5 concludes the paper with a discussion on the GLAV approach.

2 Framework for Data Integration

We set up a formal framework for data integration in the relational setting. We assume that the databases involved in our framework are defined over a fixed (infinite) set Δ of objects. A database $\mathcal{DB}$ for a relational schema $\mathcal{R}$ is a relational structure $(\Delta^{\mathcal{DB}}, \cdot^{\mathcal{DB}})$ over $\mathcal{R}$ with $\Delta^{\mathcal{DB}} \subseteq \Delta$. When needed, we denote

a relation r of arity n by r/n. Given a query q over $\mathcal{DB}$, we denote by $q^{\mathcal{DB}}$ the set of tuples of objects in $\Delta^{\mathcal{DB}}$ obtained by evaluating q over $\mathcal{DB}$, i.e., the set of *answers* to q over $\mathcal{DB}$. In particular, we focus on *conjunctive queries* (CQs) with equality atoms and constants. We denote a CQ of arity n over a relational schema $\mathcal{R}$ as

$$\{ \langle X_1, \ldots, X_n \rangle \mid \varphi(X_1, \ldots, X_n, Y_1, \ldots, Y_m) \}$$

where $X_1, \ldots, X_n$ are the *distinguished variables* (not necessarily pairwise distinct), $Y_1, \ldots, Y_m$ are the existentially quantified *non-distinguished variables*, and $\varphi(X_1, \ldots, X_n, Y_1, \ldots, Y_m)$ is a conjunction of atoms over predicate symbols in $\mathcal{R}$, involving constants, and the variables $X_1, \ldots, X_n, Y_1, \ldots, Y_m$. For a relation r/n, we write the CQ $\{\langle X_1, \ldots, X_n \rangle \mid r(X_1, \ldots, X_n)\}$ simply as r.

We consider also constraints over a relational schema. In particular, we consider inclusion dependencies, simple equality-generating dependencies, and single head full dependencies [2]. Given a relation r and a tuple $\mathbf{A}$ of distinct attributes of r, we denote the projection of r over $\mathbf{A}$ by $r[\mathbf{A}]$. Similarly, given a tuple t of r, we denote the projection of t over $\mathbf{A}$ by $t[\mathbf{A}]$. An *inclusion dependency* is a dependency of the form $r[\mathbf{A}] \subseteq r'[\mathbf{A}']$, where r and r' are two relations of a relational schema $\mathcal{R}$ and $\mathbf{A}$ and $\mathbf{A}'$ are two sequences of distinct attributes of the same arity, belonging to r and r' respectively. A database $\mathcal{DB}$ satisfies $r[\mathbf{A}] \subseteq r'[\mathbf{A}']$ if $r[\mathbf{A}]^{\mathcal{DB}} \subseteq r'[\mathbf{A}']^{\mathcal{DB}}$. A *simple equality-generating dependency* has the form $r \rightarrow A = A'$, where r is a relation of a relational schema $\mathcal{R}$, and A and A' are two distinct attributes of r. A database $\mathcal{DB}$ satisfies $r \rightarrow A = A'$ if for every tuple $t \in r^{\mathcal{DB}}$, it holds that $t[A] = t[A']$. A *single head full dependency* has the form $q \subseteq r$, where r is a relation of a relational schema $\mathcal{R}$ and q is a conjunctive query over $\mathcal{R}$ of the same arity as r. A database $\mathcal{DB}$ satisfies $q \subseteq r$ if $q^{\mathcal{DB}} \subseteq r^{\mathcal{DB}}$.

A *data integration system* $\mathcal{I}$ is a triple $\langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$, where

- $\mathcal{G}$ is the *global schema*, expressed in the relational model, possibly with constraints.
- $\mathcal{S}$ is the *source schema*, also expressed in the relational model.
- $\mathcal{M}$ is the *mapping* between $\mathcal{G}$ and $\mathcal{S}$, constituted by a set of *assertions* of the form

  $$q_{\mathcal{S}} \subseteq q_{\mathcal{G}}$$

  where $q_{\mathcal{S}}$ and $q_{\mathcal{G}}$ are two queries of the same arity, respectively over the source schema $\mathcal{S}$ and over the global schema $\mathcal{G}$.

Intuitively, the source schema describes the schema of the data sources, which contain data, while the global schema provides a reconciled, integrated, view of the underlying sources. The assertions in the mapping establish the connection between the relations of the global schema and those of the source schema. As typical in data integration, we consider here mappings that are *sound*, i.e., the data provided by the queries over the sources satisfy the queries over the global schema, but do not necessarily characterize completely the answer of the queries over the global schema [16, 9, 7]. User queries are posed over the global

schema and are answered by retrieving data from the sources, making use of the mapping.

Two basic approaches for specifying the mapping have been proposed in the literature: *global-as-view* (GAV) and *local-as-view* (LAV) [16, 9]. In the GAV approach, the mapping $\mathcal{M}$ associates to each relation g in $\mathcal{G}$ a query $\varrho_{\mathcal{S}}(g)$ over $\mathcal{S}$, i.e., a GAV mapping is a set of assertions, one for each relation g of $\mathcal{G}$, of the form

$$\varrho_{\mathcal{S}}(g) \subseteq g$$

In the LAV approach, the mapping $\mathcal{M}$ associates to each relation s in $\mathcal{S}$ a query $\varrho_{\mathcal{G}}(s)$ over $\mathcal{G}$, i.e., a LAV mapping is a set of assertions, one for each relation s of $\mathcal{S}$, of the form

$$s \subseteq \varrho_{\mathcal{G}}(s)$$

Observe that in both cases we associate to a relation (either global or local) a single query. We call *GAV (with constraints)* the class of integration systems (with constraints) with a GAV mapping. Similarly for *LAV (with constraints)*.

Given an integration system $\mathcal{I} = \langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$, we call *source database* (for $\mathcal{I}$), a database for the source schema $\mathcal{S}$, and *global database* (for $\mathcal{I}$) a database for $\mathcal{G}$ satisfying the constraints of $\mathcal{G}$. Let $\mathcal{D}$ be a source database. A global database $\mathcal{B}$ *satisfies an assertion* $q_{\mathcal{S}} \subseteq q_{\mathcal{G}}$ *in* $\mathcal{M}$ *with respect to* $\mathcal{D}$, if $q_{\mathcal{S}}^{\mathcal{D}} \subseteq q_{\mathcal{G}}^{\mathcal{B}}$. The global database $\mathcal{B}$ is said to be *legal for* $\mathcal{I}$ *with respect to* $\mathcal{D}$, if it satisfies all assertions in the mapping $\mathcal{M}$ with respect to $\mathcal{D}$. Observe that, in general, several global databases exist that are legal for $\mathcal{I}$ with respect to $\mathcal{D}$.

Queries posed to an integration system $\mathcal{I}$ are expressed in terms of the relations in the global schema of $\mathcal{I}$. Given a source database $\mathcal{D}$ for $\mathcal{I}$, the answer $q^{\mathcal{I},\mathcal{D}}$ to a query q to $\mathcal{I}$ with respect to $\mathcal{D}$, is the set of tuples t of objects in $\mathcal{D}$ such that $t \in q^{\mathcal{B}}$ for *every* global database $\mathcal{B}$ legal for $\mathcal{I}$ with respect to $\mathcal{D}$. The set $q^{\mathcal{I},\mathcal{D}}$ is called the set of *certain answers* of q to $\mathcal{I}$ with respect to $\mathcal{D}$.

Given two integration systems $\mathcal{I} = \langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$ and $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ over the same source schema $\mathcal{S}$ and such that all relations of $\mathcal{G}$ are also relations of $\mathcal{G}'$, we say that $\mathcal{I}'$ is *query-preserving* with respect to $\mathcal{I}$, if for every query q to $\mathcal{I}$ and for every source databases $\mathcal{D}$ for $\mathcal{S}$, we have that

$$q^{\mathcal{I},\mathcal{D}} = q^{\mathcal{I}',\mathcal{D}}$$

In other words, we say that $\mathcal{I}'$ is query-preserving with respect to $\mathcal{I}$ if, given a query over the global schema of $\mathcal{I}$, the certain answers we get for the query on the two integration systems are identical.

To compare classes of integration systems, we introduce the concept of query-reducibility. A class $\mathcal{C}_1$ of integration systems is *query-reducible* to a class $\mathcal{C}_2$ of integration systems if there exist a function $f : \mathcal{C}_1 \to \mathcal{C}_2$ such that, for each $\mathcal{I}_1 \in \mathcal{C}_1$ we have that $f(\mathcal{I}_1)$ is query-preserving with respect to $\mathcal{I}_1$.

3 Comparing LAV and GAV without Constraints

In this section we consider data integration systems without constraints in the global schema. We want to check whether any GAV system can be transformed

into a LAV one which is query-preserving wrt it, and vice-versa. We show that both transformation are not feasible.

We begin with the transformation from LAV to GAV.

Theorem 1. *The class of LAV data integration systems is not query-reducible to the class of GAV systems.*

Proof. We prove the theorem by exhibiting a particular LAV system $\mathcal{I} = \langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$, a source database $\mathcal{D}$ for $\mathcal{S}$, and a set of queries such that, for any GAV system $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$, the certain answers of the queries wrt $\mathcal{D}$ differ in $\mathcal{I}$ and $\mathcal{I}'$.

The LAV system $\mathcal{I}$ is as follows. The global schema $\mathcal{G}$ is constituted by $g_1/2$ and $g_2/2$, while the source schema $\mathcal{S}$ is constituted by a single relation $s/2$. The mapping $\mathcal{M}$ is

$$\varrho_{\mathcal{G}}(s) \;=\; \{\langle X, Y \rangle \mid g_1(X, Z) \wedge g_2(Z, Y)\}$$

By contradiction, assume there is a GAV system $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ that is query-preserving with respect to $\mathcal{I}$. Observe that, since no constraints are allowed in the global schema, the introduction of a new relation in $\mathcal{G}'$ is useless if we want to construct a system that is query-preserving wrt $\mathcal{I}$; in fact, the newly introduced predicates could not be related to g_1 and g_2. Therefore, we can assume that $\mathcal{G}' = \mathcal{G}$. It follows that the mapping $\mathcal{M}'$ has the form

$$\varrho_{\mathcal{S}}(g_1) = \{\langle X, Y \rangle \mid \xi_1(X, Y, Z_1, \ldots, Z_{k_1}, c_1, \ldots, c_{h_1})\}$$
$$\varrho_{\mathcal{S}}(g_2) = \{\langle X, Y \rangle \mid \xi_2(X, Y, W_1, \ldots, W_{k_2}, d_1, \ldots, d_{h_2})\}$$

where ξ_1 and ξ_2 are conjunctions of atoms over the only relation s, $Z_1, \ldots, Z_{k_1}$ and $W_1, \ldots, W_{k_2}$ are existentially quantified variables, and $c_1, \ldots, c_{h_1}$ and $d_1, \ldots, d_{h_2}$ are constants of Δ.

We take the source database $\mathcal{D}$ to be such that $s^{\mathcal{D}} = \{\langle a, b \rangle\}$, where a and b are two constants, and we consider the following queries:

$$q_1(X, Y) = \{\langle X, Y \rangle \mid g_1(X, Z) \wedge g_2(Z, Y)\}$$
$$q_2(X, Y) = \{\langle X, Y \rangle \mid g_1(X, Y)\}$$
$$q_3(X, Y) = \{\langle X, Y \rangle \mid g_2(X, Y)\}$$

The certain answers of q_1, q_2, and q_3 to $\mathcal{I}$ wrt $\mathcal{D}$ are the following: $q_1^{\mathcal{I},\mathcal{D}} = \langle a, b \rangle$, $q_2^{\mathcal{I},\mathcal{D}} = \emptyset$, and $q_3^{\mathcal{I},\mathcal{D}} = \emptyset$.

If one of $\varrho_{\mathcal{S}}(g_1)^{\mathcal{D}}$ or $\varrho_{\mathcal{S}}(g_2)^{\mathcal{D}}$ is non-empty, we have that one of $q_2^{\mathcal{I}',D}$ or $q_3^{\mathcal{I}',D}$ is non-empty, and hence a contradiction. When both $\varrho_{\mathcal{S}}(g_1)^{\mathcal{D}}$ and $\varrho_{\mathcal{S}}(g_2)^{\mathcal{D}}$ are empty, we immediately obtain that $q_1^{\mathcal{I}',\mathcal{D}} = \emptyset$. Contradiction.

This result shows that the mechanism of query answering in LAV cannot be directly simulated by the corresponding mechanism in GAV, which is basically unfolding, i.e., the substitution in the user query of the global relations with their definition given by the mapping.

We now turn to the transformation from GAV to LAV.

Theorem 2. *The class of GAV data integration systems is not query-reducible to the class of LAV systems.*

Proof. We exhibit a particular GAV system $\mathcal{I} = \langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$ and a query such that, for any LAV system $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ we can construct a source database $\mathcal{D}$ for $\mathcal{S}$ such that the certain answers of the query to $\mathcal{I}$ and $\mathcal{I}'$ differ wrt $\mathcal{D}$.

Let $\mathcal{I}$ be as follows. The global schema $\mathcal{G}$ is constituted by a single relation $g/2$, while the source schema $\mathcal{S}$ is constituted by $s_1/2$ and $s_2/2$. The mapping $\mathcal{M}$ is

$$\varrho_{\mathcal{S}}(g) \;=\; \{\langle X, Y\rangle \mid s_1(X, Z) \wedge s_2(Z, Y)\}$$

As in the previous case, we observe that the introduction of new relations in $\mathcal{G}'$ is not significant if we want to construct a system that is query-preserving wrt $\mathcal{I}$. Hence we assume that $\mathcal{G}' = \mathcal{G}$, and the mapping $\mathcal{M}'$ has the form

$$\varrho_{\mathcal{G}}(s_1) = \{\langle X, Y\rangle \mid \eta_1(X, Y, Z_1, \ldots, Z_{k_1}, c_1, \ldots, c_{h_1})\}$$
$$\varrho_{\mathcal{G}}(s_2) = \{\langle X, Y\rangle \mid \eta_2(X, Y, W_1, \ldots, W_{k_2}, d_1, \ldots, d_{h_2})\}$$

where η_1 and η_2 are conjunctions of atoms over the only relation g, $Z_1, \ldots, Z_{k_1}$, $W_1, \ldots, W_{k_2}$ are existentially quantified variables, and $c_1, \ldots, c_{h_1}, d_1, \ldots, d_{h_2}$ are constants in Δ.

We define the source database $\mathcal{D}$ such that $s_1^{\mathcal{D}} = \{\langle a, b\rangle\}$ and $s_2^{\mathcal{D}} = \{\langle b, c\rangle\}$, where a, b, and c are constants, distinct from $c_1, \ldots, c_{h_1}, d_1, \ldots, d_{h_2}$. Consider the query

$$q(X, Y) \;=\; \{\langle X, Y\rangle \mid g(X, Y)\}$$

whose certain answers in $\mathcal{I}$ are $\{\langle a, c\rangle\}$.

Let $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ be a LAV system. We show that $\langle a, c\rangle \notin q^{\mathcal{I}', \mathcal{D}}$, by constructing a global database $\mathcal{B}'$ which satisfies $\mathcal{M}'$ wrt $\mathcal{D}$ and such that $\langle a, c\rangle \notin q^{\mathcal{B}'}$. We construct $g^{\mathcal{B}'}$ as follows. We associate to each variable or constant V appearing in the definition of $\varrho_{\mathcal{S}}(s_1)$ a distinct constant $\psi(V)$, such that $\psi(X) = a$, $\psi(Y) = b$, and $\psi(V) = V$ if V is a constant. Then, for each atom $g(V_1, V_2)$ appearing in $\varrho_{\mathcal{S}}(s_1)$, we add the tuple $\langle \psi(V_1), \psi(V_2)\rangle$ to $g^{\mathcal{B}'}$. We do the same for $\varrho_{\mathcal{S}}(s_2)$, with $\psi(X) = b$ and $\psi(Y) = c$. Such a construction of $g^{\mathcal{B}'}$ ensures that $\langle a, c\rangle \notin g^{\mathcal{B}'}$ (by construction) and that $\mathcal{B}'$ is legal for $\mathcal{I}'$ wrt $\mathcal{D}$, as $\langle a, b\rangle \in s_1^{\mathcal{B}'}$ and $\langle b, c\rangle \in s_2^{\mathcal{B}'}$. Therefore $\langle a, c\rangle \notin q^{\mathcal{I}', \mathcal{D}}$. This proves the claim.

This result shows that we are not able to deduce the information of a LAV mapping, which specifies the role of each source relation wrt the global schema, from the information contained in a corresponding GAV mapping, which gives direct information on how query answering may be performed.

4 Comparing LAV and GAV with Constraints

We address the question of query-reducibility in the case where integrity constraints are allowed in the global schema.

One direction is almost immediate: single head full dependencies suffice for query-reducibility from GAV systems to LAV systems. Indeed, if $\mathcal{I} = \langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$ is a GAV system, we define a corresponding LAV system $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ as follows. For every source relation s in $\mathcal{S}$, we define a corresponding new relation g_s in $\mathcal{G}'$, and we include in $\mathcal{M}'$ the assertion $s \subseteq \varrho_{\mathcal{G}}(g_s)$. Now, for every $\varrho_{\mathcal{S}}(g) \subseteq g$ in $\mathcal{M}$, we introduce in $\mathcal{G}'$ the single head full dependency $\rho'_{\mathcal{S}}(g) \subseteq g$, where $\rho'_{\mathcal{S}}(g)$ denotes the conjunction obtained from $\rho_{\mathcal{S}}(g)$ by substituting every atom $s(x_1, \ldots, x_n)$ with $g_s(x_1, \ldots, x_n)$. It is easy to see that the resulting data integration system $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ is a LAV system, and that the transformation is query-preserving. Observe also that the size of $\mathcal{I}'$ is linearly related to the size of $\mathcal{I}$.

We now turn to the question of reducing LAV systems to GAV systems. We show that, when inclusion and simple equality generating dependencies are allowed on the global schema, we can obtain from every LAV system a query-preserving GAV system. Let $\mathcal{I} = \langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$ be a LAV integration system. Without loss of generality, we can assume that no equality atoms appear in the conjunctive queries in the mapping $\mathcal{M}$. We define a corresponding GAV integration system $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ as follows. For technical reasons, we first rewrite all queries in the mapping $\mathcal{M}$ so that variables appear in each atom at most once, by adding suitable equalities to the body of the queries. For example, the query $\{\langle X \rangle \mid \mathsf{cites}(X, X)\}$ is rewritten as $\{\langle X \rangle \mid \mathsf{cites}(X, Y) \wedge Y = X\}$.

Then $\mathcal{I}$ is as follows:

- The set of sources $\mathcal{S}$ remains unchanged.
- The global schema $\mathcal{G}'$ is obtained from $\mathcal{G}$ by introducing:
  - a new relation $\mathit{image\_s}/n$ for each relation s/n in $\mathcal{S}$;
  - a new relation $\mathit{expand\_s}/(n+m)$ for each relation s/n in $\mathcal{S}$, where m is the number of non-distinguished variables of $\varrho_{\mathcal{G}}(s)$; we assume variables in $\varrho_{\mathcal{G}}(s)$ to be enumerated as $Z_1, \ldots, Z_{n+m}$, with $Z_1, \ldots, Z_n$ being the distinguished variables;

  and by adding the following dependencies:
  - for each relation s/n in $\mathcal{S}$ we add the inclusion dependency

$$\mathit{image\_s}[1, \ldots, n] \subseteq \mathit{expand\_s}[1, \ldots, n]$$

  - for each relation s in $\mathcal{S}$ and for each atom $g(Z_{i_1}, \ldots, Z_{i_k})$ occurring in $\varrho_{\mathcal{G}}(s)$, we add the inclusion dependency

$$\mathit{expand\_s}[i_1, \ldots, i_k] \subseteq g[1, \ldots, k]$$

  - for each relation s in $\mathcal{S}$ and for each atom $Z_i = Z_j$ occurring in $\varrho_{\mathcal{G}}(s)$, we add the simple equality generating dependency

$$\mathit{expand\_s} \rightarrow i = j$$

- The GAV mapping $\mathcal{M}'$ associates to each global relation $\mathit{image\_s}$ the query

$$\varrho_{\mathcal{S}}(\mathit{image\_s}) = s$$

  and to the remaining global relations the empty query.

It is immediate to verify the following theorem.

Theorem 3. *Let $\mathcal{I}$ be a LAV integration system, and $\mathcal{I}'$ the corresponding GAV integration system defined as above. Then $\mathcal{I}'$ can be constructed in time that is linear in the size of $\mathcal{I}$.*

We illustrate the transformation with an example.

Example 1. Consider a LAV integration system $\mathcal{I} = \langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$ where:

- The global schema $\mathcal{G}$ is constituted by the relations $\mathsf{cites}/2$, expressing that a paper cites another paper, and $\mathsf{sameTopic}/2$, expressing that two papers are on the same topic.
- The source schema $\mathcal{S}$ is constituted by three relations: source_1, containing pairs of papers that mutually cite each other; source_2, containing pairs of papers on the same topic, each with at least one citation; and source_3, containing papers that cite themselves.
- The LAV mapping $\mathcal{M}$ between the source schema and the global schema is:

$$\begin{aligned}
&\varrho_{\mathcal{G}}(\mathsf{source}_1) = \{\langle X, Y\rangle \mid \mathsf{cites}(X, Y) \wedge \mathsf{cites}(Y, X)\} \\
&\varrho_{\mathcal{G}}(\mathsf{source}_2) = \{\langle X, Y\rangle \mid \mathsf{sameTopic}(X, Y) \wedge \mathsf{cites}(X, Z) \wedge \mathsf{cites}(Y, W)\} \\
&\varrho_{\mathcal{G}}(\mathsf{source}_3) = \{\langle X\rangle \mid \mathsf{cites}(X, Y) \wedge X = Y\}
\end{aligned}$$

Then the corresponding GAV integration system $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ is as follows:

- The source schema $\mathcal{S}$ remains unchanged.
- The global schema $\mathcal{G}'$ is constituted by the relations $\mathsf{cites}/2$, $\mathsf{sameTopic}/2$ as before, and the additional relations $\mathit{image}\_\mathsf{source}_1/2$, $\mathit{image}\_\mathsf{source}_2/2$, $\mathit{image}\_\mathsf{source}_3/1$, $\mathit{expand}\_\mathsf{source}_1/2$, $\mathit{expand}\_\mathsf{source}_2/4$, and $\mathit{expand}\_\mathsf{source}_3/2$. Moreover, $\mathcal{G}$ contains the following inclusion dependencies:

$$\begin{aligned}
&\mathit{image}_\mathsf{source}_1[1,2] \subseteq \mathit{expand}_\mathsf{source}_1[1,2] \\
&\mathit{image}_\mathsf{source}_2[1,2] \subseteq \mathit{expand}_\mathsf{source}_2[1,2] \\
&\mathit{image}_\mathsf{source}_3[1] \subseteq \mathit{expand}_\mathsf{source}_3[1] \\
&\mathit{expand}_\mathsf{source}_1[1,2] \subseteq \mathsf{cites}[1,2] \\
&\mathit{expand}_\mathsf{source}_1[2,1] \subseteq \mathsf{cites}[1,2] \\
&\mathit{expand}_\mathsf{source}_2[1,3] \subseteq \mathsf{cites}[1,2] \\
&\mathit{expand}_\mathsf{source}_2[2,4] \subseteq \mathsf{cites}[1,2] \\
&\mathit{expand}_\mathsf{source}_3[1,2] \subseteq \mathsf{cites}[1,2] \\
&\mathit{expand}_\mathsf{source}_2[1,2] \subseteq \mathsf{sameTopic}[1,2] \\
&\mathit{expand}_\mathsf{source}_3 \rightarrow 1 = 2
\end{aligned}$$

- The GAV mapping $\mathcal{M}'$ is

$$\varrho_{\mathcal{S}}(\mathit{image}\_\mathsf{source}_i) = \mathsf{source}_i, \qquad i \in \{1, 2, 3\}$$

■

We now show that the LAV integration system $\mathcal{I}$ and the corresponding GAV integration system $\mathcal{I}'$ obtained as above are indeed query-equivalent. The proof is based on the observation that both integration systems $\mathcal{I}$ and $\mathcal{I}'$ can be captured by suitable *logic programs* (we refer to [13] for notions relative to logic programming).

We first concentrate on GAV systems. The logic program $\mathcal{P}_{\mathcal{I}'}$ associated to a GAV system $\mathcal{I}' = \langle \mathcal{G}', \mathcal{S}, \mathcal{M}' \rangle$ is defined as follows:

- For each inclusion dependency $g_1[\mathbf{A}] \subseteq g_2[\mathbf{B}]$ in $\mathcal{G}'$, where $\mathbf{A}$ and $\mathbf{B}$ are sets of attributes, we first introduce a "pseudo-rule" of the form (assuming for simplicity that the attributes in $\mathbf{A}$ and $\mathbf{B}$ are the first h ones in g_1 and g_2, respectively):

$$g_2(X_1, \ldots, X_h, X_{h+1}, \ldots, X_n) \leftarrow g_1(X_1, \ldots, X_h, Y_{h+1}, \ldots, Y_m)$$

  Then, for each simple equality generating dependency in $\mathcal{G}$ of the form $g_2 \rightarrow i{=}j$, we substitute in the above pseudo-rule each occurrence of X_j with X_i. We skolemize the resulting pseudo-rule, obtaining a rule of the form

$$g_2(Z_1, \ldots, Z_k, f_{k+1}(Z_1, \ldots, Z_k), \ldots, f_n(Z_1, \ldots, Z_k)) \leftarrow g_1(Z_1, \ldots, Z_k, W_{k+1} \ldots, W_m)$$

  where each f_i is a fresh Skolem function.
- For each assertion $\varrho_{\mathcal{S}}(g) \subseteq g$ in the mapping $\mathcal{M}'$, where $\varrho_{\mathcal{S}}(g) = \{\langle X_1, \ldots, X_n \rangle \mid \varphi(X_1, \ldots, X_n, Y_{n+1}, \ldots, Y_m)\}$, we have a rule of the form

$$g(X_1, \ldots, X_n) \leftarrow \varphi(X_1, \ldots, X_n, Y_{n+1}, \ldots, Y_m)$$

  with the proviso that, if a simple equality generating dependency applies to g, then we have to equate the appropriate variables.

In addition, the relations in $\mathcal{S}$ can be seen as predicates that are given extensionally. That is, a source database $\mathcal{D}$ for $\mathcal{I}'$ can be seen as a finite set of ground facts in logic programming terms.

By applying results from logic programming theory [13], we can show the following lemma.

Lemma 1. *Let $\mathcal{I}'$ be a GAV integration system, $\mathcal{D}$ a source database for $\mathcal{I}'$, $\mathcal{P}_{\mathcal{I}'}$ the corresponding logic program as defined above, and M_{min} the minimal model of $\mathcal{P}_{\mathcal{I}'} \cup \mathcal{D}$. Then, given a query q over $\mathcal{G}'$, for every tuple $\langle c_1, \ldots, c_n \rangle$ of objects in $\mathcal{D}$ we have that*

$$\langle c_1, \ldots, c_n \rangle \in q^{\mathcal{I},\mathcal{D}} \quad \textit{if and only if} \quad \langle c_1, \ldots, c_n \rangle \in q^{M_{min}}$$

Proof (sketch). By considering the semantics of constraints in $\mathcal{G}'$, and the corresponding translation in $\mathcal{P}_{\mathcal{I}'}$, it can be shown that the certain answers of q to $\mathcal{I}'$ wrt $\mathcal{D}$ are those that are correct answers to q for the logic program $\mathcal{P}_{\mathcal{I}'} \cup \mathcal{D}$. The claim follows from the classical result in logic programming that the correct answers to a logic program are those that are true in the minimal model. □

In other words, for GAV integration systems, the tuples of constants in the certain answer to a query q are equal to those that satisfy q in the minimal model of the corresponding logic program.

Let us turn to LAV integration systems. Without loss of generality, we can assume that equality generating dependencies have been folded into queries by suitably renaming variables. Given a LAV integration system $\mathcal{I}$, we can define an associated logic program $\mathcal{P}_\mathcal{I}$ by introducing rules for dependencies as before, and by treating queries in the mapping as done in [5]. In particular, given the query associated to source s (for simplicity of presentation, we assume s to be a unary relation and the relations in the query to be binary)

$$\varrho_\mathcal{G}(s) \;=\; \{\langle X\rangle \mid g_1(X, Y_1) \wedge \cdots \wedge g_k(X, Y_k)\}$$

by applying skolemization we get

$$\varrho_\mathcal{G}(s) \;=\; \{\langle X\rangle \mid g_1(X, f_1(X)) \wedge \cdots \wedge g_k(X, f_k(X))\}.$$

Then, we can introduce in $\mathcal{P}_\mathcal{I}$ the following rules, derived from the skolemized query:

$$\begin{gathered} g_1(X, f_1(X)) \leftarrow s(X) \\ \cdots \\ g_k(X, f_k(X)) \leftarrow s(X) \end{gathered}$$

Based on the results in [5], we can prove also for LAV integration systems a lemma analogous to Lemma 1.

Lemma 2. *Let $\mathcal{I}$ be a LAV integration system, $\mathcal{D}$ a source database for $\mathcal{I}$, $\mathcal{P}_\mathcal{I}$ the corresponding logic program as defined above, and M_{min} the minimal model of $\mathcal{P}_\mathcal{I} \cup \mathcal{D}$. Then, given a query q over $\mathcal{G}$, for every tuple $\langle c_1, \ldots, c_n\rangle$ of objects in $\mathcal{D}$ we have that*

$$\langle c_1, \ldots, c_n\rangle \in q^{\mathcal{I},\mathcal{D}} \qquad \textit{if and only if} \qquad \langle c_1, \ldots, c_n\rangle \in q^{M_{min}}$$

In other words, also for LAV integration systems, the tuples of constants in the certain answer to a query q are equal to those that satisfy q in the minimal model of the corresponding logic program.

With these lemmas in place we can prove our main result.

Theorem 4. *Let $\mathcal{I}$ be a LAV integration system, and $\mathcal{I}'$ the corresponding GAV integration system defined as above. Then $\mathcal{I}'$ is query-preserving wrt $\mathcal{I}$.*

Proof (sketch). Let $\mathcal{P}_\mathcal{I}$ be the logic program capturing $\mathcal{I}$ and $\mathcal{P}_{\mathcal{I}'}$ the logic program capturing $\mathcal{I}'$. Then it is possible to show that, for every source database $\mathcal{D}$ for $\mathcal{I}$ and every global relation g of the global schema $\mathcal{G}$ of $\mathcal{I}$, we have (modulo renaming of the Skolem functions) that

$$g^{M_{min}} = g^{M'_{min}}$$

where M_{min} and M'_{min} are the minimal model of $\mathcal{P}_\mathcal{I} \cup \mathcal{D}$ and of $\mathcal{P}_{\mathcal{I}'} \cup \mathcal{D}$, respectively. Hence, by considering Lemma 1 and Lemma 2, we get the claim. □

5 Discussion

In the previous sections we have studied the relative expressive power of the two main approaches to data integration, namely, LAV and GAV. We have shown that, in the case where integrity constraints are not allowed in the global schema, LAV and GAV systems are not mutually query-reducible. On the other hand, the presence of integrity constraints allows us to derive query-preserving transformations in both directions.

In particular, we have demonstrated that inclusion dependencies and a simple form of equality-generating dependencies in the global schema are sufficient for transforming any LAV systems into a query-preserving GAV system whose size is linearly related to the size of the original system. Interestingly, the technique can be easily extended for transforming any GLAV system into a GAV one.

In the GLAV approach to data integration, the relationships between the global schema and the sources are established by making use of both LAV and GAV assertions [6]. More precisely, in a GLAV system, we associate a conjunctive query $q_{\mathcal{G}}$ over the global schema to a conjunctive query $q_{\mathcal{S}}$ over the source schema. Therefore, GLAV generalizes both LAV and GAV.

By exploiting the technique presented in Section 4, it is not difficult to see that any GLAV system can be transformed into a query-preserving GAV one, with the same technique presented above. The key idea is that a GLAV assertion can be transformed into a GAV assertion plus an inclusion dependency. Indeed, for each assertion

$$q_{\mathcal{S}} \subseteq q_{\mathcal{G}}$$

in the GLAV system (where the arity of both queries is n), we introduce a new relation symbol r/n in the global schema of the resulting GAV system, and we associate to r the query

$$\varrho_{\mathcal{S}}(r) = q_{\mathcal{S}}$$

plus the inclusion

$$r \subseteq q_{\mathcal{S}}$$

Now, it is immediate to verify that the above inclusion can be treated exactly with the same technique introduced in the LAV to GAV transformation, and therefore, from the GLAV system we can obtain a query-preserving GAV system whose size is linearly related to the size of the original system.

References

[1] Serge Abiteboul and Oliver Duschka. Complexity of answering queries using materialized views. In *Proc. of the 17th ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS'98)*, pages 254–265, 1998. 339

[2] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases.* Addison Wesley Publ. Co., Reading, Massachussetts, 1995. 341

[3] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Moshe Y. Vardi. Answering regular path queries using views. In *Proc. of the 16th IEEE Int. Conf. on Data Engineering (ICDE 2000)*, pages 389–398, 2000. 339

[4] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Moshe Y. Vardi. View-based query processing and constraint satisfaction. In *Proc. of the 15th IEEE Symp. on Logic in Computer Science (LICS 2000)*, pages 361–371, 2000. 339

[5] Oliver M. Duschka and Michael R. Genesereth. Answering recursive queries using views. In *Proc. of the 16th ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS'97)*, pages 109–116, 1997. 348

[6] Marc Friedman, Alon Levy, and Todd Millstein. Navigational plans for data integration. In *Proc. of the 16th Nat. Conf. on Artificial Intelligence (AAAI'99)*, pages 67–73. AAAI Press/The MIT Press, 1999. 349

[7] Gösta Grahne and Alberto O. Mendelzon. Tableau techniques for querying information sources through global schemas. In *Proc. of the 7th Int. Conf. on Database Theory (ICDT'99)*, volume 1540 of *Lecture Notes in Computer Science*, pages 332–347. Springer, 1999. 339, 341

[8] Jarek Gryz. Query folding with inclusion dependencies. In *Proc. of the 14th IEEE Int. Conf. on Data Engineering (ICDE'98)*, pages 126–133, 1998. 339

[9] Alon Y. Halevy. Answering queries using views: A survey. *Very Large Database J.*, 10(4):270–294, 2001. 338, 339, 341, 342

[10] Richard Hull. Managing semantic heterogeneity in databases: A theoretical perspective. In *Proc. of the 16th ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS'97)*, 1997. 339

[11] Alon Y. Levy. Logic-based techniques in data integration. In Jack Minker, editor, *Logic Based Artificial Intelligence*. Kluwer Academic Publisher, 2000. 339

[12] Chen Li and Edward Chang. Query planning with limited source capabilities. In *Proc. of the 16th IEEE Int. Conf. on Data Engineering (ICDE 2000)*, pages 401–412, 2000. 339

[13] John W. Lloyd. *Foundations of Logic Programming (Second, Extended Edition)*. Springer, Berlin, Heidelberg, 1987. 347

[14] Lucian Popa, Alin Deutsch, Arnaud Sahuguet, and Val Tannen. A chase too far? In *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, pages 273–284, 2000. 340

[15] Xiaolei Qian. Query folding. In *Proc. of the 12th IEEE Int. Conf. on Data Engineering (ICDE'96)*, pages 48–55, 1996. 339

[16] Jeffrey D. Ullman. Information integration using logical views. In *Proc. of the 6th Int. Conf. on Database Theory (ICDT'97)*, volume 1186 of *Lecture Notes in Computer Science*, pages 19–40. Springer, 1997. 339, 341, 342

[17] Ron van der Meyden. Logical approaches to incomplete information. In Jan Chomicki and Günter Saake, editors, *Logics for Databases and Information Systems*, pages 307–356. Kluwer Academic Publisher, 1998. 339