COMPLETENESS IN THE RELATIONAL MODEL: A COMPREHENSIVE FRAMEWORK

(Research Paper)

Monica Scannapieco
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Roma, Italy
monscan@dis.uniroma1.it

Carlo Batini
Dipartimento di Informatica, Sistemistica e Comunicazione
Università di Milano "Bicocca"
Milano, Italy
batini@disco.unimib.it

Abstract: Completeness is a well known data quality dimension in the area of databases. Intuitively, a database is complete if it represents every fact of the real world coherent with the database semantics, i.e. its intension. In the paper, we provide a comprehensive framework for characterizing completeness in the relational model, investigating several different paradigms typical of database models, such as closed world and open world assumptions, and presence or absence of null values. Furthermore, we introduce an algebra for completeness, in order to address the problem of calculating composition of quality dimensions in queries that include relational operators such as union, difference and cartesian product. Under different assumptions and for the different types of completeness, we provide properties and shortcuts for such an algebra.

Key Words: Completeness, Algebra, Composition of Quality Dimensions

1 INTRODUCTION

Data quality can be defined by a set of *dimensions*, also called quality properties or characteristics (e.g. [1], [7], [9]); examples of such dimensions are completeness, accuracy, consistency and currency.

When data have an associated quality and go through manipulations of different types, it is important to understand which is the quality of the data resulting from such manipulations. In order to be able to calculate the resulting quality, there is the general need to investigate an *algebra* for data quality dimensions.

The general problem statement for the definition of a data quality algebra is represented in Figure 1. In the figure, the data **X**, described according to a data model *M*, are processed by a generic composition function **F** on the set of operators $o_1 \ldots o_r$ associated with *M*. Also, a function $\mathbf{Q_D}$ calculates the value of the quality dimension *D* for **X** and for **Y=F(X)**. We aim to define, if it exists, the function $\mathbf{Q^F{}_D}$ that calculates the value of the quality dimension *D* for **Y** starting from the value of the quality dimension *D* for **X**, instead of calculating such a value directly on **Y** by applying the function $\mathbf{Q_D}$.

$$\begin{tikzcd}
X \arrow[r, "F"] \arrow[d, "Q_D"'] & Y=F(X) \arrow[d, "Q_D"] \\
Q_D(X) \arrow[r, "Q^F_D?"] & Q_D(Y)
\end{tikzcd}$$

Figure 1: Defining an algebra for data quality dimensions

In this paper we will consider the particular case of this problem in which:

- *M* is the relational model;
- $o_1 \ldots o_r$ correspond to the relational operators *union, intersection, cartesian product*;
- *D* is a specific data quality dimension, i.e. **completeness**. Intuitively, a database is complete if it represents every fact of the real world coherent with the database semantics;
- Q^F_D is a function that evaluates the completeness of the elements of the relational model under different hypotheses and for different relational operators.

In real scenarios, especially when data are replicated across different sources, such as in Cooperative Information Systems [4], it is usual to obtain new data by combining data sets, extracted from one or more sources. In such contexts, it is important to be able to calculate the completeness of the new resulting data starting from the completeness values of the original sources. Furthermore, in order to enhance completeness, it is often not enough to consider single sources and engage improvement actions on them independently; instead, such actions should be properly complemented by composing data from different sources. We believe that in both cases, namely querying for retrieving higher completeness and engaging completeness improvement actions, an algebra supporting composition operations is strongly needed.

Our investigation considers as a reference scenario the set of public administrations that cooperate each other in an e-Government scenario. There are many incompleteness problems that arise in this context. Such problems are due to uncontrolled data acquisition processes that, for instance, do not define mandatory fields, thus allowing to enter any type of information whether complete or not. As another example, due to missing data entries or missing data exchanges in a batch update session, it may happen that significant amounts of data are not entered at all. Furthermore, update processes do not often work in the proper way. For instance, the data entry of just born citizens are often not performed with the correct periodicity with respect to the frequency of births, thus causing public databases to be affected by incompleteness problems systematically. Besides trying to properly re-engineer such processes, a different strategy concerns getting as much information as possible from the whole set of administrations: by combining many incomplete, totally or partially replicated data, a higher level of completeness could be obtained.

A practical scenario that shows our idea is the set of population registries that are managed by public agencies. As an example, in Italy for each city council there are : (i) a personal data registry for resident people and, (ii) a separate registry for civil status of resident people. At the regional level, there are the local income tax payers registry, and a separate social insurance registry. All these sources have their own completeness. In many administrative processes these sources are combined, and it would be of interest to understand the completeness of the result of the combination starting from the source completeness. Let us also notice that sources that represent the same reality may have different completeness values, due to erroneous update and insert processes. For instance, in the Italian public administration there are two different registries of Italian citizens living abroad, whose sizes are respectively 3.5 millions and 4.5 millions of citizens, corresponding to different related completeness values. Such registries are owned by different agencies, with different processes managing data; ongoing activities are now trying to merge the two registries.

This paper will be mainly related to formally describe the different concepts of completeness that may arise in the relational model, when different assumptions are made. Secondly, we will show how an algebra for completeness can be introduced on the basis of the provided set of definitions; in the paper we will consider one single case, leaving to future work the extension to other meaningful cases. More specifically, the organization of the paper is as follows. In Section 2, some basic considerations will be introduced, in order to describe the proposed framework for completeness. In Sections 3 and 4 the detail of the provided definitions will be presented. In Section 5, the first steps towards an algebra for completeness are provided. Finally, Section 6 draws the related work and Section 7 summarizes the results of the paper and describes the future work.

2 A Framework for Completeness in the Relational Model

In this section we will first introduce the notation that will be used throughout the paper. Then, we will describe the framework we propose to extensively define completeness in the relational model.

2.1 Preliminaries on the Relational Model

A domain *Dom* is a set of constants. $\boldsymbol{D}$ is a finite set of domains. The cartesian product Dom_1 *x* Dom_2 *x,…, x* Dom_n is the set of elements $d_1,d_2,\ldots,d_n$, where $d_i \in Dom_i$. A *relation r* is a finite subset of the product of one or more domains. If $r \subseteq Dom_1$ *x* Dom_2 *x,…, x* Dom_n, each element of *r* has the form $d_1,d_2,\ldots,d_n$ and is called a *tuple* of *r*. The *cardinality* of *r* is the number of tuples in it. Each d_i is said to be a *component* of the tuple. The name of the component is called an *attribute*.

We assume a set $\boldsymbol{A}$ of attributes. Attributes are associated to domains through a mapping $\boldsymbol{Map: A \rightarrow D}$ such that given an attribute $A \in \boldsymbol{A}$, ***Map***(A) is a domain $Dom \in \boldsymbol{D}$ called *domain* of *A*. A set *S* of attributes is called *relation schema*. Given a relation $r \subseteq Dom_1$ *x* Dom_2 *x,…, x* Dom_n and its relation schema $S= A_1,A_2,\ldots,A_n$, a set of attributes $K \in S$ that uniquely designate each tuple of *r* is called a *key* of *r*.

We assume the relational algebra operators: *union*, *intersection*, *cartesian product*. The choice of these operators is motivated by the aim of composing completeness of relational elements, as detailed in the following of the paper.

2.2 The Framework

We consider two classification coordinates that define a space for the characterization of the completeness dimension in the relational model.

The first classification coordinate considers the possibility of two assumptions in the relational model, namely: the Closed World Assumption (CWA) and the Open World Assumption (OWA). The CWA [8] assumes, in the context of a logical formulation of the relational model, that the tuples in a relation are all and only the tuples that satisfy the relational schema. On the contrary, the OWA assumes that the tuples in the relation are a subset of the tuples satisfying the relational schema.

The second classification coordinate considers the presence/absence of null values in the relational model. By combining the presence/absence of null values with the OWA and CWA assumption, we obtain in principle four different models, that are shown in Figure 2:

1. CWA without null;
2. OWA without null;
3. CWA with null;
4. OWA with null.

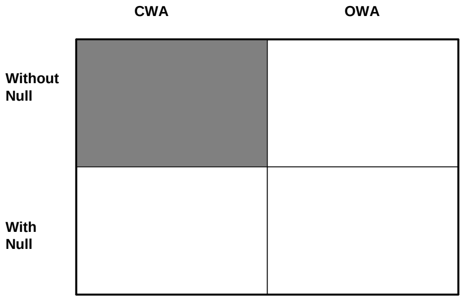

Figure 2: A framework for completeness in the relational model

In case 1, the concept of completeness is not meaningful. Incompleteness problems arise when data are missing, therefore no incompleteness problem will occur when all tuples are represented, like in the CWA model, and all related attribute values are present, like in the absence of the null values hypothesis.

The remaining three cases correspond to meaningful cases in real world situations, and require different definitions for completeness that will be detailed in the following sections.

3 COMPLETENESS IN THE OWA MODEL WITHOUT NULL VALUES

When considering the OWA model, in order to characterize completeness, we need to introduce the concept of a *reference relation*.

Definition 1. Reference Relation of *r***.** *The reference relation of r, called ref(r), is the relation containing all tuples that satisfies the relational schema of r.*

As an example, if *dept* is a relation representing the employees of a given department, and a given employee of the department is not represented as a tuple of *r*, then the tuple corresponding to the missing employee is in *ref(r)*. In practical cases, the reference relations are rarely available, instead their cardinality is much easier to get. There are also cases in which the reference relation is available but only periodically (e.g. when a census is performed).

Under the OWA model without null values, we define completeness as the fraction of tuples actually represented in a relation *r*, with respect to the whole number of tuples in *ref(r)*. More formally:

Definition 2.Completeness of *r***.** *The completeness of r, called C(r), is:*

$$C(r) = \frac{cardinality\ of\ r}{cardinality\ of\ ref(r)}$$

Example 1. Let us consider all the citizens of Rome. From the personal registry of Rome's city council, the overall number is 6 millions. Let us suppose that a company stores Rome's citizens for the purpose of its business. The cardinality of the relation *r* storing Rome's citizens is 5.400.000. The completeness of *r*, *C(r)*, is equal to 0.9.

In the next section more complex notions of completeness will be investigated.

4 Completeness in the OWA and CWA Models with Null Values

Different meanings for null values in the relational model have been proposed, including *unknown* and *inapplicable* [2]. In this paper, we will assume to extend the domains of all attributes of a relational schema with the *unknown* null meaning. Other meanings for null values will be introduced in our framework in future work.

We provide definitions for completeness considering three different axes, namely:

1. Type of the world considered, i.e. OWA vs CWA.
2. Granularity of the model elements, i.e. value, tuple, attribute and relations, as shown in Figure 3.
3. Strictness, i.e. a weak or strong characterization depending on evaluating completeness as a percentage or as a boolean function respectively.

Specifically, considering both the OWA and the CWA models, we define:

- *value completeness* to capture the presence of null values for some attributes of tuples;
- *tuple completeness* to characterize the completeness of a whole tuple with respect to the values of all attributes; the tuple completeness is further distinguished into *weak tuple completeness* and *strong tuple completeness*;
- *attribute completeness* to measure the number of null values of a specific attribute in a relation. Also in this case we will see that both a *weak attribute completeness* and a *strong attribute completeness* are relevant;
- *relation completeness* that captures the presence of null values in the whole relation. A *weak relation completeness* and a *strong relation completeness* are again defined.

The following definitions will be provided for a relation *r* of a relational schema *S*= $A_1,A_2,\ldots,A_n$.

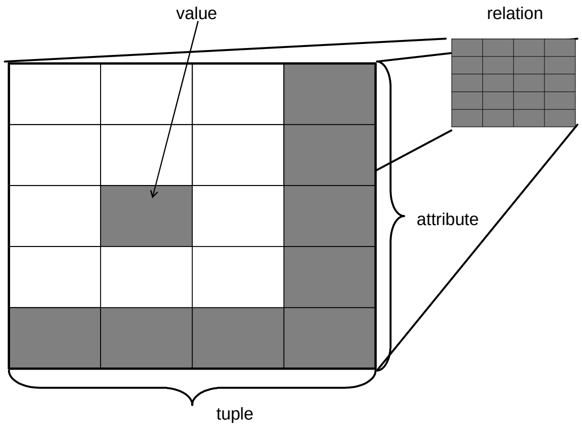

Figure 3: Completeness of different elements of the relational model

Definition 3. Value Completeness-Completeness of a tuple t **wrt an Attribute A_i.** *Given an attribute A_i and a tuple $t = d_1,d_2,\ldots,d_n$, we define the completeness of t with respect to A_i, namely C_t^{Ai} as follows:*

$C_t^{A_i} = 0$ if $d_i = null$

$C_t^{A_i} = 1$ otherwise

This is the most common type of completeness that considers the presence or the absence of a specific value of an attribute in a tuple.

Definition 4. Weak Tuple Completeness. *Given a tuple $t = d_1, d_2, \ldots, d_n$, we define the weak tuple completeness $C_w(t)$ as follows:*

$$C_w(t) = \sum_{i=1}^{i=n} \frac{C_t^{A_i}}{n}$$

Example 2. The weak tuple completeness evaluates the percentage of specified values in the tuple with respect to the total number of attributes of the tuple itself. One way to see the weak tuple completeness is as a measure of the information content carried on by the tuple with respect to the maximum potential information content of the tuple. With reference to this interpretation, we are implicitly assuming that all values of the tuple equally contribute to the total information content of the tuple. In a relation representing persons with their personal data, including, for instance, SSN, NAME, SURNAME, AGE, EYECOLOR, a tuple with some missing values can be still significant, e.g. for the EYECOLOR attribute, and such a measure points out the degree of the actual information content of the tuple.

Definition 5. Strong Tuple Completeness. *Given a tuple $t = d_1, d_2, \ldots, d_n$, we define the strong tuple completeness $C_s(t)$ as follows:*

$C_s(t) = 1$ if $C_t^{A_i} = 1, \forall i \in [1..n]$

$C_s(t) = 0$ otherwise

Example 3. The strong tuple completeness is useful in all the applications in which it is not possible to admit null values in a tuple. Let us assume, for instance, to consider a relational schema representing addresses and consisting of STREET NAME, CIVIC NUMBER and CITY; let us also assume that an application needs to ship letters to such addresses. In this case, the strong tuple completeness is more suitable than the weak tuple completeness as no delivery can be performed with partial addresses (assuming that the delivery is guaranteed only by the presence of all attribute values).

Definition 6. Weak Attribute Completeness. *Given an attribute A_i and given a generic tuple t_j , we define the weak attribute completeness of A_i , namely $C_w(A_i)$, as follows:*

$$C_w(A_i) = \sum_{j=1}^{card(r)} \frac{C_{t_j}^{A_i}}{card(r)}$$

Example 4. Let us consider an application calculating the average of votes obtained by students from a relation storing STUDENT-NUMBER and VOTE. The absence of some values for the VOTE attribute simply implies a deviation in the average calculus but does not preclude the calculus itself.

Definition 7. Strong Attribute Completeness. *Given an attribute A_i and given a generic tuple t_j, we define the strong attribute completeness of A_i , namely $C_s(A_i)$, as follows:*

$$C_s(A_i) = 1 \qquad \text{if } C_{t_j}^{A_i} = 1, \forall j \in [1..card(r)]$$

$$C_s(A_i) = 0 \qquad \text{otherwise}$$

Example 5. Let us consider an application for air traffic control. It could not be admitted at all the absence of even a single value for the an attribute AIRPLANE-POSITION, therefore a strong characterization of completeness is required.

Definition 8. Weak Relation Completeness. *Given the relation r, we define the weak relation completeness of r, namely $C_w(r)$, as follows:*

$$C_w(r) = \sum_{j=1}^{card(r)} \frac{C_w(t_j)}{card(r)} = \frac{\sum_{j=1}^{card(r)} \sum_{i=1}^{n} \frac{C_{t_j}^{A_j}}{n}}{card(r)}$$

Example 6. The weak relation completeness is relevant in all the applications that need to evaluate the completeness of a whole relation and can admit the presence of null values on some attributes. For instance, given a relation representing the students of a university: (i) in the CWA model, the weak relation completeness measures how much information is represented by the relation, evaluating the actually available information content; (ii) in the OWA model, the weak relation completeness takes into account also the possibility of missing students in the relation so providing even a finer granularity measure of the information content of the relation.

Definition 9. Strong Relation Completeness. *Given the relation r, we define the strong relation completeness of r , namely $C_s(r)$,' as follows:*

$$C_s(r) = 1 \qquad \text{if } C_s(t_j) = 1, \forall j \in [1..card(r)]$$

$$C_s(r) = 0 \qquad \text{otherwise}$$

Example 7. The strong relation completeness is useful whereas no tuple is admitted to have any null value. For instance, let us consider a relation representing the students of a university course together with their email addresses. Let us also suppose that an application sending notifications of examination dates is designed such that either all students are notified at the same time, or none of them receives notifications. This implies that a strong relation completeness evaluation is needed. Then, assuming the CWA model, notifications are sent when emails of all represented students are present; conversely, assuming the OWA model, notifications are sent when all the actual students are represented with their emails too.

While value completeness and tuple completeness are valid in both the OWA and CWA models, attribute completeness and relation completeness have different definitions in the two models.
For the sake of conciseness, the definitions we have provided for attribute completeness and for relation completeness, both in the weak and strong cases, are proposed within the CWA model. Nevertheless, they can be easily adapted to the OWA model on the basis of the following property.

Property 1. Attribute and Relation Completeness in the OWA Model. The definitions of attribute completeness and relation completeness in the OWA model, both in the weak and strong cases, are such that:

- if *card(ref(r))=card(r),* they coincide with the definitions provided in the CWA case;
- if *card(ref(r))>card(r),* then the definitions are the same as the ones provided in the CWA case but replacing *card(r)* with *card(ref(r)).*

5 TOWARDS AN ALGEBRA FOR COMPLETENESS IN THE RELATIONAL MODEL

In the previous sections, we have provided a reference set of definitions that fully characterizes the completeness in the relational model.

In this section, we start showing how such definitions are useful in order to introduce an algebra for completeness in the relational model. More specifically, we start considering the completeness definition provided in Section 3, i.e. the OWA model without null values, and we consider the completeness evaluation for some operators of the relational algebra. We are especially interested in the operators composing two different relations, therefore we will consider: union, intersection and cartesian product. We assume to have the following input parameters:

- r_1 and r_2 relations;
- $card(r_1)$ and $card(r_2)$ cardinalities;
- $card(ref(r_1))$ and $card(ref(r_2))$ cardinalities.

Notice that we do not assume to have the reference relations themselves, but only their cardinalities.

5.1 Case 1: Union

Given the two relations r_1 and r_2, let r be equal to $r_1 \cup r_2$. We consider how to compute $C(r)$ from $C(r_1)$ and $C(r_2)$. We consider the following subcases.

Case 1.1: Same Reference Relation. We suppose that $ref(r_1)=ref(r_2)=ref(r)$. In the case in which no additional knowledge on relations is available, it is easy to show that the completeness of r is such that:

$$C(r) \geq \ max(C(r_1), C(r_2))$$

Behind this inequality, $C(r)$ can be better characterized if some intensional knowledge on tuples contained in r_1 and r_2 is available. Specifically, we can distinguish three further cases:

- If $r_1 \cap r_2 = \varnothing$ then $C(r)=C(r_1)+C(r_2)$
- If $r_1 \cap r_2 \neq \varnothing$ then $C(r)>max(C(r_1),C(r_2))$
- If $r_1 \cap r \neq \varnothing$ and $r_1 \subseteq r_2$ then C(r)=C(r₂)

Example 8. In Figures 4(a) and 4(b), the two relations *dept1* and *dept2* are shown, each representing professors of a university department and having the reference relation, *r-dept=ref(dept1)=ref(dept2)* corresponding to all the professors of the department. We have the following input data:
cardinality of *dept1* is 4, cardinality of *dept2* is 5, cardinality of *r-dept* is 8. Hence, C(*dept1*) is 0.5 and *C(dept2)* is 0.625.
From this information we can derive:
C(*dept1* ∪ *dept2*) >= 0.625
In Figure 4(c), the relation *dept3* is shown, the cardinality of which is 4; notice that this relation only contains associate professors, therefore $dept1 \cap dept3 = \varnothing$. In this case, we can easily compute:
$C(dept1 \cap dept3)$=0.5+0.5=1

In Figure 4(d), the relation *dept4* is shown, the cardinality of which is 2; notice that $dept4 \subseteq dept1$.
In this case, we have: $C(dept1 \cap dept4)$=0.5

Case 1.2: Different Reference Relation. We consider a case that often occurs in real scenarios, i.e. when reference relations are a disjoint and complete partition of a domain. More specifically, we suppose that $ref(r_1) \cap ref(r_2) = \emptyset$ *and* $ref(r) = ref(r_1) \cup ref(r_2)$.
In this case, it is easy to show that the completeness of *r* is:

$$C(r) = \frac{card(r_1)+card(r_2)}{card(ref(r_1))+card(ref(r_2)} = \frac{C(r_1)*card(r_1)+C(r_2)*card(r_2)}{card(ref(r_1))+card(ref(r_2))}$$

ID	Surname	Name	Role
1	Rossi	Marco	Full
2	Terzi	Enrico	Full
3	Damiano	Claudia	Full
4	Bianchi	Federico	Full

(a) dept1

ID	Surname	Name	Role
1	Ambretta	Toni	Associate
2	Terzi	Enrico	Full
3	Damiano	Claudia	Full
4	Micelli	Monia	Associate
5	Bianchi	Federico	Full

(b) dept2

ID	Surname	Name	Role
1	Ambretta	Toni	Associate
2	Ultimo	Franco	Associate
3	Rieti	Angela	Associate
4	Giorgi	Maria	Associate

(c) dept3

ID	Surname	Name	Role
1	Rossi	Marco	Full
2	Damiano	Claudia	Full

(d) dept4

Figure 4: Examples of completeness evaluation from the union of two relations

5.2 Case 2: Intersection

Given the two relations r_1 and r_2, let *r* be equal to $r_1 \cap r_2$. We consider how to compute *C(r)* from $C(r_1)$ and $C(r_2)$. We consider the only interesting case in which the reference relation $ref(r_1)$ and $ref(r_2)$ are different.
In this case, we have:

$$C(r) = \frac{card(r)}{card(ref(r))} = \frac{card(r)}{card((ref(r_1) \bigcap ref(r_2)))} =$$
$$= \frac{card(r)}{card(ref(r_1)) + card(ref(r_2)) - card(ref(r_1) \bigcup (ref(r_2)))}$$

If card(*ref(r$_1$)∪ref(r$_2$))=0* then *card(r)=0 and C(r)=0;*
otherwise.

$$C(r) > \frac{card(r)}{card(ref(r_1)) + card(ref(r_2))}$$
$$= card(r) * \frac{C(r_1) * C(r_2)}{C(r_1) * card(r_2) + C(r_2) * card(r_1)}$$

Example 9. Let us consider again the relation *dept2* shown in Figure 4(b), the completeness of which is $C(dept_2) = \frac{card(dept_2)}{card(ref(dept_2))} = \frac{5}{8} = 0.625$

Let us consider the relation *instr*, containing professors of different departments that have been instructors for a company. The relation is shown in Figure 5, from which we derive that *card(instr)*=3. Let us assume it is known that only the 50% of instructors has been stored in the relation *instr*, i.e. *card(ref(instr))=6*; it follows that *C(instr)*=0.5. Let us consider again the relation *dept2*, shown in Figure
4(b), and representing professors of a university department. If we are interested to understand which professors of such a department have been instructors for the company to which *instr* is related, we have to intersect *dept2* and *instr*. We can find a lower bound for this intersection as:

$$C(dept_2) = \frac{card(dept_2)}{card(ref(dept_2))} = \frac{5}{8} = 0.625$$

5.3 Case 3: Cartesian Product

Given the relations r_1 and r_2, let *r* be equal to r_1× r_2., i.e. *r* is the cartesian product of r_1 and r_2.
In this case, it is easy to compute:

$$C(r) = \frac{card(r_1 \times r_2)}{card(ref(r_1) \times ref(r_2))} = \frac{card(r_1) * card(r_2)}{card(ref(r_1)) * card(ref(r_2))} =$$
$$C(r_1) * C(r_2)$$

ID	Surname	Name	Role
1	Ambretta	Toni	Associate
2	Bagnoli	Carlo	Full
3	Damiano	Claudia	Full

Figure 5: The relation *instr*

6 Related Work

The problem of defining an algebra for data quality dimensions has been already considered in the literature. We have selected four proposals that, according to our knowledge, explicitly deal with such a problem, namely: Motro 98 [3], Wang 01[10], Parssian 02 [6] e Naumann 04 [5]. In Figure 6, these works are compared to our proposal on the basis of : (i) the adopted model, (ii) the chosen quality dimensions, (iii) the goal of the proposal, (iv) the criteria to evaluate the chosen quality dimensions, and (v) the algebra operators taken into account.

With reference to the goal, in our approach, we aim to compose different relations in order to come out with an enriched information content; only Naumann 04 has a similar approach, aiming at integrating data sources. Conversely, the other approaches are more focused on querying to obtain quality values, indeed they are less focused on compositional operators. We can summarize the originality of our approach as follows:

- OWA and reference relation. All cited works consider the open world assumption. Also, the reference relation is a concept already present in some works; for instance in [10], the concept of *real world state*, which is very close to the reference relation concept, is introduced. In our case, we make two different assumptions on the reference relations of the two relations involved in the composition: (i) the two reference relations are the same, i.e. the corresponding schemas are the same, and, as a consequence, the composition is performed over the same reality of interest; (ii) the two reference relations are different, therefore the interpretation of the composition is that the two schemas are different too, and so the composition results in an integration of the two schemas.
- With respect to Motro 98, we assume to have sound relations and the completeness criteria introduced in Motro 98 corresponds to our completeness in the OWA without null. Therefore, we introduce much more criteria for completeness.
- In Wang 01, a different quality dimension is considered, namely accuracy. With respect to the algebra operators a probabilistic approach is adopted in Wang 01, while we consider a deterministic definition of algebraic relational operators for completeness. We also differ for providing a richer classification of completeness with respect to different models.
- In Parssian 02, like in Motro 98, completeness is defined only at the granularity of relations, by relying on motivations concerning the cost of computing finer quality profiles. We have provided many examples that instead justify the need of a deeper characterization of completeness.
- In Naumann 04, some defined completeness criteria correspond to our criteria; for instance the coverage corresponds to our OWA without null relation completeness and the density of an attribute corresponds to CWA with null weak attribute completeness. With respect to Naumann's work, we have chosen to provide a richer characterization of completeness by introducing a strictness criteria, i.e. strong or weak, and introducing also a tuple-level completeness. Furthermore, we have chosen to stay within the relational model, without introducing new operators, like merge operators. Indeed, our aim is to study completeness, as well as other quality dimensions, within the relational theory; this allows us to rely on well-studied research result such as the investigation on the CWA and OWA models, and on different semantics for null values.

7 CONCLUDING REMARKS

In this paper, we have described a framework for defining completeness in the relational model. We have also discussed how the framework is useful to define an algebra for completeness, by focusing on one case. Our principal goals for the future work are: (i) to complete the proposed algebra in order to include also further provided completeness definitions, (ii) to consider the algebra also with different null values meanings. As already sketched in the Introduction, we will also investigate an algebra for further quality dimensions. The aim of our study is indeed to define the compositional fragment of the relational algebra, namely union, intersection and cartesian product, also for other quality dimensions.

	Model	**Quality Dimensions**	**Goal**	**Algebra Operators**	**Criteria**
Motro 1998	Relational model with OWA (implicit)	Soundness Completeness	- Estimating the quality of query results - Information sources selection	- Cartesian Product - Selection - Projection	- Soundness - Completeness
Wang 2001	Relational model with OWA (implicit)	Accuracy	- Estimating the quality of query results	- Selection - Projection - Cartesian Product	- Deterministic and Probablistic Tuple Accuracy - Relation Accuracy - Probabilistic Attribute Accuracy - Null Relational Accuracy
Parssian 2002	Relational model with OWA (implicit)	Accuracy Completeness	- Estimating the quality of query results	- Selection - Projection - Cartesian Product - Join	- Relation Accuracy - Relation Inaccuracy - Relation Mismembership - Relation Incompleteness
Naumann 2004	Set of data sources With OWA and CWA (implicit)	Completeness	- Sources composition - Information sources selection	- Join Merge - Full Outerjoin Merge - Left Outerjoin Merge - Right Outerjoin Merge	- Coverage - Attribute Density - Source Density - Query Dependant Density - Completeness
Our Approach	Relational model with OWA and CWA	Completeness	- Relational sources composition - Relational sources selection	- Union - Intersection - Cartesian Product	- Relation Completeness (OWA without Null) - Value Completeness (with Null) - Attribute completeness (With Null, OWA+CWA, strong+weak) - Tuple completeness (With Null, OWA+CWA, strong+weak) - Relation Completeness (With Null, OWA+CWA, strong+weak)

Figure 6: Comparison with other approaches to define a data quality algebra

Finally, we are interested to investigate how a wide range of definitions for data quality dimensions can influence and modify database design methodologies. More specifically, methodological guidelines need to be introduced in order to choose the more suitable model for a specific application, among different possible models (e.g. OWA vs. CWA). Furthermore, logical design of data and applications has to be enriched with activities that allow monitoring and achieving the desired data quality dimensions levels.

Acknowledgements

This work has been supported by MIUR through the FIRB 2001 project MAIS, http://www.mais-project.it. It has been also supported by the FISR 2002 project "Le nuove tecniche di governo e l'innovazione tecnologica nella formazione e attuazione delle politiche pubbliche".

REFERENCES

[1] Bovee, M., Srivastava, R:P. and Mak, B.R., A Conceptual Framework and Belief-Function Approach to Assessing Overall Information Quality, in Proceedings of the *6th International Conference on Information Quality*, Boston, MA, 2001.

[2] Imielinski, T. and Lipski, W., "Incomplete Information in Relational Database", *Journal of the Association for Computing Machinery*, 31(4) 1984, pp.761-791.

[3] Motro, A. and Ragov, I., Estimating Quality of database, in Proceedings of the *3rd International Conference on Flexible Query Answering Systems*, Roskilde, Denmark, 1998.

[4] Mylopoulos, J. and Papazoglou, M.P. (eds.), Cooperative Information Systems (Special Issue), *IEEE Expert Intelligent Systems & Their Applications, 12*(5), 1997.

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

[6] Parssian, A., Sarkar, S. and Jacob, V.S., Assessing Information Quality for the Composite Relational Operation Join, in Proceedings of the *7th International Conference on Information Quality*, Boston, MA, 2002.

[7] Redman, T.C., *Data Quality for the Information Age*, Artech House, 1996.

[8] Reiter, R., "On Closed World Databases", *Logic and Databases* (H.Gallaire and J. Minker, eds.), Plenum Press, 1978.

[9] Wang, R.Y. , "A Product Perspective on Total Data Quality Management", *Communications of the ACM*, 41(2) 1998.

[10] Wang, R.Y., Ziad, M. and Lee, Y.W. , *Data Quality*, Kluwer AcademicPublisher, 2001, pp. 63-77.