Towards a Compositional Semantic Account of Data Quality Attributes

Lei Jiang[1], Alex Borgida[1,2], and John Mylopoulos[1,3]

[1] University of Toronto
[2] Rutgers University
[3] University of Trento
{leijiang,jm}@cs.toronto.edu, borgida@cs.rutgers.edu

Abstract. We address the fundamental question: what does it mean for data in a database to be of high quality? We motivate our discussion with examples, where traditional views on data quality are found to be unsatisfactory. Our work is founded on the premise that data values are primarily linguistic signs that convey meaning from their producer to their user through senses and referents. In this setting, data quality issues arise when discrepancies occur during this communication. We sketch a theory of senses for individual values in a relational table based on its semantics expressed using some ontology. We use this to offer a compositional approach, where data quality is expressed in terms of a variety of primitive relationships among values and their senses. We evaluate our approach by accounting for quality attributes in other frameworks proposed in the literature. This exercise allows us to (i) reveal and differentiate multiple, sometimes conflicting, definitions of a quality attribute, (ii) accommodate competing views on how these attributes are related, and (iii) point to possible new definitions.

1 Introduction

The quality of any artifact is determined by the degree to which it fulfills its intended use ("fitness for purpose"). Arguably, for a database the purpose is answering questions about the application it models. Data quality (DQ), the fitness of data values for question-answering purposes, is widely accepted as a multi-dimensional and a hierarchical concept [23,13,3]. More than a dozen proposals have been made to characterize and define various aspects of DQ (also called *quality dimensions* or *quality attributes*) in terms of a classification scheme. Examples of such schemes include (i) accessibility, interpretability, usefulness and believability DQ [23] (ii) intrinsic, contextual, representational, and accessibility DQ [24], and (iii) mandatory vs. desirable, primary vs. secondary, and direct vs. indirect DQ [5].

Criticism of these approaches includes ambiguity, subjectiveness, and even circularity of definitions within a single classification [4], and inconsistency across multiple classifications [13]. As an example of circular definition, credibility in [23] is considered as a sub-attribute of believability, but it is itself defined as having sufficient evidence to be believed; as an example of inconsistent definition, in [24] completeness and believability belong to two disjoint categories, while they are related

Q. Li et al. (Eds.): ER 2008, LNCS 5231, pp. 55–68, 2008.

through a specialization link in [23]. This lack of precision and consistency in defining DQ attributes also prevents one from answering even the most basic questions about how DQ attributes relate. For example, does imprecision imply inaccuracy? Does our judgment of completeness presuppose the notion of relevance? Do concepts such as trust, believability and credibility refer to the same DQ attribute? If not, how do they differ?

The objective of this paper is to address these problems by offering a formal framework for DQ. In particular, we consider a DQ attribute as a complex expression, where the meaning of the attribute is captured in terms of the meaning of its constituents and the structure of the expression. Instead of defining each DQ attribute separately, we seek to answer the following questions: (i) what are the primitive constituents from which DQ attributes can be expressed and (ii) how can these constituents be combined in a meaningful way. The concept of "*sign*" provides such a primitive notion for the investigation of these questions. Data values in a database are above all linguistic signs that convey meaning from their producer to their user; DQ issues arise when discrepancies occur during this communication. Based on these observations, we propose a novel, compositional approach to understand and define DQ attributes in terms of a variety of primitive relationships between values and their senses. We evaluate our approach by accounting for DQ attributes in other frameworks proposed in the literature. This exercise allows us to (i) reveal and differentiate multiple, sometimes conflicting, definitions of a quality attribute; (ii) accommodate competing views on how these attributes should be related; and (iii) point to possible new definitions.

The rest of paper is structured as follows. We motivate our discussion with some examples where traditional views on DQ are unsatisfactory in determining whether data is defective (Section 2). We then describe our view of data quality based on a triadic model of signs (Section 3), and sketch a theory of senses for individual values in a relational table based on its semantics expressed using some ontology (Section 4). Next, we present the compositional approach to DQ and its evaluation (Section 5 and 6). Finally, we review related work (Section 7), and concluded and point to our future research plan (Section 8).

2 Motivating Examples

Consider a *Patient* table (Table 1) that records body temperatures for patients in a hospital. Suppose that each row shown here records the temperature of a particular patient at different time points (other rows are omitted). First, let us consider accuracy, one of the most studied DQ attributes. It has been defined as a measure of "*the closeness between a value v and a value v', considered as the correct representation of the real-life phenomenon v aims to represent*" [22,3]. For example, if the patient's real name is *v' = 'Ben Cheung'*, but was recorded as *v = 'Ben Franklin'* instead, we may conclude that *v* is inaccurate.

Example 1. In some cases, our judgment of accuracy does not rely on syntactic proximity of data values, but is affected instead by our interpretation of their meanings. For example, it would have been no less accurate to have *'98.6°F'* instead of *'37.0°C'* in the last row, as long as we understand that these two values represent the same temperature reading using different scales.

Table 1. The Patient table

Name	Temperature	Time
Ben Cheung	37.2°C	2007/11/05 13:05
Ben Cheung	38.5°C	2007/11/06 12:00
Ben Cheung	37.0°C	2007/11/07 11:55

Example 2. Moreover, whether a data value is considered accurate often depends on both its interpreted and intended meaning. For example, if there is no agreement on how the temperature should to be measured, we may interpret *'37.2°C'* in the first row as Ben's temperature measured under normal conditions, while it really represents his temperature after aspirin was administered. Inaccuracy caused by such a mismatch cause no less a problem than a typographical error (e.g., entering *'36.2°C'* instead of *'37.2°C'*).

Example 3. Furthermore, accuracy cannot be considered in isolation: our judgment on accuracy of a value depends on the judgment of that of its related values. For example, consider *'38.5°C'* and *'2007/11/06 12:00'* in the second row. If we know that Ben's temperature was 39 degree Celsius on Nov. 6, 2007 at 12:00, we may want to conclude that *'38.5°C'* represents the real-world phenomenon (i.e., 39 degree Celsius) inaccurately. But, in doing so we have already made an assumption that *'2007/11/06 12:00'* is accurate! What if we instead know that Ben's temperature was 38.5 degree Celsius on Nov. 6, 2007 at 11:45? In this case, are we willing to believe that it is the time not the temperature value that was inaccurately recorded?

Consider next completeness, another commonly studied DQ attribute, which has been defined as the percentage of all tuples satisfying the relational schema of a table (i.e., tuples in the true extension of the schema) which are actually presented in the table [3].

Example 4. Actually, it is impossible to talk about the "true" extension of a relational schema without knowing what the user's requirements are. Accordingly, the above data about Ben Cheung could be complete or incomplete depending on whether Ben's temperature is required to be measured only once or twice a day.

3 Nature of Data Quality

In this section, we describe our view of DQ, founded on the notion of signs [17]. Generally speaking, a *sign* is something that stands to someone for something else. Accordingly, we see values (together with their metadata) in databases as primarily linguistic signs standing for real world phenomena. Information processing is a form of communication realized by creating, passing and utilizing signs [12]; DQ issues arise when discrepancies occur during this communication.

In the *meaning triad* [12], a triadic sign model, a *symbol* (e.g., *'Ben Cheung'*) is connected to a *referent* (e.g., a particular person in the world), and a *sense* understood by its interpreter (e.g., the concept of that person in the interpreter's mind). The difference between the referent and sense of a symbol could be understood in analogy to that of the extensional and intensional definitions of a term. Moreover a symbol may have more than one "valid" sense (and referent), under different circumstances, according to different interpreters.

We find it useful to distinguish four kinds of senses/referents of a symbol:

- The *intended sense/referent* is the sense/referent of the symbol according to its *producer*. It is the meaning the producer intends to communicate, and is determined exclusively by the producer.
- The *interpreted sense/referent* is the sense/referent of the symbol according to its *user*. It is the meaning the user recognizes, and is determined exclusively by the user.
- The *supposed sense/referent* is the sense/referent, determined exclusively by the *requirements for production* of the symbol, such as conventions and regulations the producer has to comply with, ethical and social norms, etc.
- The *expected sense/referent* is the sense/referent, determined exclusively by the *conditions for use* of the symbol, such as the tasks, purposes and goals of the user.

To illustrate this distinction, consider the temperature value *'37.2°C'* in Table 1. Suppose Sudha, the doctor of Ben, needs to know his temperature, not lowered by an antipyretic, and measured around noon every day (because he is plotting a graph with X-axis points every 24 hours). She also expects the measurement to be taken using a thermometer in the mouth. A new nurse, Catherine, running late, measured Ben's temperature at 13:05, with a thermometer in the ear. Moreover, Catherine is unaware of the fact that Ben had taken an antipyretic at 12:40. As a result, by recording *'37.2°C'*, Catherine *intended* to say "Ben's temperature without antipyretic, measured at 13:05 with a tympanal thermometer". If Catherine had been more careful, this value's *supposed* meaning would be "Ben's temperature after antipyretic, measured at 13:05 with some thermometer". On the other hand, Sudha may *interpret* this value as "Ben's temperature without antipyretic, measured at 13:05 (because he saw the time value in the table) with an oral thermometer", which is different from what he *expected*: "Ben's temperature without antipyretic, measured around noon with an oral thermometer".

Ideally, total data quality means that the four types of senses must match for each data value individually, and certain constraints must hold among the same types of senses for related values, especially ones in different fields of the same row. DQ issues arise when this does not hold. For example, when Sudha expects oral measurements, but this requirement is not specified explicitly, discrepancy is likely to exist between the expected and supposed senses. More generally, if some sources of variability (e.g., the type of thermometer used and patient conditions) are not captured in the data (or metadata), the communication between the producer and user will be ambiguous. Of course, whether or not such ambiguity is considered problematic depends on the purpose for which the data is to be used, and it is the role of the requirements specification to eliminate these problems.

4 Nature of Senses

Before using the preceding distinctions in a theory of DQ, it helps to flesh out a bit the notion of "sense" we have in mind. In this paper we concentrate on data values concerning object properties (e.g., length, temperature and color), rather than general relationships between objects. For this purpose, we follow the DOLCE ontology [14] in viewing the world as populated by entities, which include concrete physical objects (e.g., persons) as well as abstract regions (e.g., distance values); the latter can appear as the values of properties[1], called *qualia*, for objects. To help communication, entities have names that allow them to be uniquely identified within some more or less restricted context: 'Ben Cheung' is presumably sufficient to identify the patient currently in the hospital in the previous example. Naming qualia allows us, for example, to have the region named 'normal temperature' contain the region named *'37°C'*, which in turn contains *'37.2°C'*. Qualia are associated with properties at specific times (which are also treated as qualia), allowing property values to change. In FOL, this might be written as ***temptrOf****('Ben Cheung', '2007/11/05 13:05') = '37.2°C'*; intensional logics use other notations [7].

The fundamental premise of databases is that one can associate a semantics with a relational table such as *Patient(NM, TPTR, TM)* along the lines of "the unique person named *NM* has temperature property value *TPTR* at time *TM*", a semantics that must be shared by data producer and user for proper communication. Given a shared ontology, this might be written in FOL as

$$\textbf{\textit{Patient}}(NM, TPTR, TM) \rightarrow \exists!p{:}\ Person\ .\ \textbf{\textit{hasName}}(p, NM) \wedge \textbf{\textit{temptrOf}}(p, TM) = TPTR$$

where we simplify matters by omitting additional variables for qualia to be "named" by *TM* and *TPTR*.

Based on this, the interpreted senses of the values in *Patient('Ben Cheung', '37.2°C', '2007/11/05 13:05')* could be *m* = "*the unique person named Ben Cheung*", *m'* = "*the temperature quale for the unique person named Ben Cheung at time quale 2007/11/05 13:05*", and *m''* = "*the time quale when the temperature quale 37.2°C was measured for the unique person named Ben Cheung*". Note that the senses of these values, and their derivation from the table semantics accounts for the situations we encountered in motivating examples in Section 2 (e.g., Example 3 concerns violation of the constraint that *m'* and *m''* must refer to the same temperature and time quale).

The above account is idealized, since it is usually necessary to observe or measure properties. This introduces a process of measurement, which allows the semantic specification to capture additional requirements. For example, the following formula specifies the kind of instrument to measure the temperature with, and a constraint on the time when measurements are to be taken:

$$\textbf{\textit{Patient}}(NM, TPTR, TM) \rightarrow \exists!p{:}\ Person, instr{:}\ OralThermometer\ .\ \textbf{\textit{hasName}}(p, NM) \wedge \textbf{\textit{measures}}(\textbf{\textit{temptrOf}}(p, TM), TPTR, instr, TM) \wedge \textbf{\textit{closeToNoon}}(TM)$$

[1] DOLCE calls properties "qualities", but we find this too confusing in our context, where we are talking about data quality. Also, DOLCE reifies properties into entities that "inhere" in objects -- a complication that is unnecessary in our context.

Moreover, measurements are almost never exact, so the precise semantics may need to talk about accuracy and precision errors for measurements or the instruments involved, the subject of metrology.

The above considerations allow us to see a basis for distinguishing different degrees of match between two senses m_1 and m_2 of a data value s, which will be important for our development of a theory of DQ. On the one hand, we have the ideal exact match $\boldsymbol{match}_{exact}(m_1, m_2)$ when the senses are identical. At the other extreme, we have a total mismatch $\boldsymbol{match}_{mismatch}(m_1, m_2)$ in cases such as when m_1 is a temperature quale while m_2 is a person. In between, we admit partial matches $\boldsymbol{match}^{attr}_{partial}(m_1, m_2)$ where *attr* is the attribute, of which *s* is a value; for example, the four senses of Ben's temperature value *'37.2°C'* discussed in the previous section would match partially. The precise details of partial match are under study, but are not important here; some of its properties include

- there is a reasoning process for deciding it, allowing for differing background knowledge (thus allowing one to discover that *"37.0°C"* and *"98.6°F"* refer to the same quale (or not));
- the arguments must agree on certain predicates and the identity of certain central entities (e.g., it is the same person's temperature that is being talked about);
- aspects concerning other predicates and entities (such as those dealing with measuring and its circumstances) are less crucial, and will lead to partial matches; the precise details of how these are to be weighted in a comparison are application goal-dependent;
- all other things being equal, the geometry of quale regions is used to compare similarity.

We also find useful a more precise variant of partial match, called $\boldsymbol{closer}^{attr}(m, m_1, m_2)$, which indicates that m_1 is conceptually closer to m than m_2 is; it allows us to find that, all other things being equal, a 13:05 measurement of a particular property is closer to a noon one than a 14:30 measurement.

5 Defining Data Quality

We characterize data quality considering four *DQ aspects*, each of which contains a collection of *theoretical DQ predicates*. These predicates are defined in terms of the relationships among symbols and their senses from a single viewpoint, therefore providing primitive constituents from which DQ attributes can be expressed. A DQ attribute in practice (e.g., accuracy, completeness) normally correspond to predicates in more than one aspect. In what follows, we discuss a few important DQ predicates in each aspect. This is, however by no means an exhaustive list of possible predicates in these aspects.

5.1 Symbol Aspect

The first DQ aspect concerns the relationships involving symbols only, without explicitly mentioning their senses. Let S be a set of symbols of interest. First we may be interested in the membership of a symbol $s \in S$ in a subset S_{accept} of S. Let us denote this using the predicate $\boldsymbol{sym}_{member}(s, S_{accept}) \Leftrightarrow s \in S_{accept}$. For example, $\boldsymbol{sym}_{member}('50°C', S_{body\text{-}temp})$ does not hold, assuming $S_{body\text{-}temp}$ is the set of symbols representing the acceptable

human body temperatures. For acceptable symbols, we may now consider a variety of relationships between them. The simplest such relationship is sameness: let $\boldsymbol{sym}_{match}(s_1, s_2)$ hold whenever s_1 and s_2 have exactly the same syntactic form. When two symbols do not match exactly, we may consider which are closer syntactically, based on some distance function $\boldsymbol{distance}_f$ (such as edit distance [3]). Let us write this using $\boldsymbol{sym}_{closer}(s, s_1, s_2) \Leftrightarrow \boldsymbol{distance}_f(s, s_1) < \boldsymbol{distance}_f(s, s_2)$. For example, $\boldsymbol{sym}_{closer}('Cheng', 'Cheung', 'Chiang')$ is true because changing from *'Cheng'* to *'Cheung'* requires fewer edits than to *'Chiang'*. Another interesting relationship, $\boldsymbol{sym}_{more\text{-}detail}(s_1, s_2)$, concerns level of detail; for real numbers we might have $\boldsymbol{sym}_{more\text{-}detail}('3.1415926', '3.14')$ indicating that, in normalized scientific notation, (i) the two arguments have the same exponent, (ii) the first argument has as least as many digits as the second one in the coefficient, and (iii) the coefficients agree in the digits presented.

5.2 The Meaning Aspect

This DQ aspect deals with the relationships involve the interpreted and intended senses of a symbol. According to H.P. Grice's classical account of speaker meaning, we rely on the *recognition* of our intention to communicate and we use that very recognition to get our message across [20]. In the context of data quality, this implies that in an ideal communication, there should be an exact match between intended and interpreted senses.

Let M be the set of senses to which the symbols in S *may* refer. First of all, we need to know whether for each symbol there is an interpreted (or intended) sense assigned to it by its user (or producer). Let us use $\boldsymbol{mea}_{has\text{-}intp}(s, m)$ (respectively, $\boldsymbol{mea}_{has\text{-}intd}(s, m)$) to indicate that a sense $m \in M$ is an interpreted (respectively, intended) sense of a symbol $s \in S$ [2]. For example, $\exists m \in M.\ \boldsymbol{mea}_{has\text{-}intp}('37.2°C', m)$ probably does not hold for a physician who doesn't work in Ben's hospital, because she will not have a way to identify the person named Ben Cheung at that hospital.

Once we know that the interpreted and intended senses exist, we can then consider whether their existence is unique. Formally, let's define

$$\boldsymbol{mea}_{has\text{-}uni\text{-}inp}(s) \Leftrightarrow \forall\, m_1, m_2 \in M.\ \boldsymbol{mea}_{has\text{-}intp}(s, m_1) \wedge \boldsymbol{mea}_{has\text{-}intp}(s, m_2) \rightarrow \boldsymbol{match}_{ex\text{-}act}(m_1, m_2),$$

$$\boldsymbol{mea}_{has\text{-}uni\text{-}int}(s) \Leftrightarrow \forall\, m_1, m_2 \in M.\ \boldsymbol{mea}_{has\text{-}intd}(s, m_1) \wedge \boldsymbol{mea}_{has\text{-}intd}(s, m_2) \rightarrow \boldsymbol{match}_{ex\text{-}act}(m_1, m_2).$$

Conversely, we may also be interested in whether two symbols are synonyms from the user's or producer's perspective (i.e., sharing their interpreted or intended senses):

$$\boldsymbol{mea}_{synonym\text{-}u}(s_1, s_2) \Leftrightarrow \exists m \in M.\ \boldsymbol{mea}_{has\text{-}intp}(s_1, m) \wedge \boldsymbol{mea}_{has\text{-}intp}(s_2, m) \wedge \neg \boldsymbol{sym}_{match}(s_1, s_2)$$

$$\boldsymbol{mea}_{synonym\text{-}p}(s_1, s_2) \Leftrightarrow \exists m \in M.\ \boldsymbol{mea}_{has\text{-}intd}(s_1, m) \wedge \boldsymbol{mea}_{has\text{-}intd}(s_2, m) \wedge \neg \boldsymbol{sym}_{match}(s_1, s_2)$$

When a symbol has an interpreted and intended sense, we are mostly interested in whether there is a match between them. First we want to know if they match exactly

$$\boldsymbol{mea}_{match}(s, m_1, m_2) \Leftrightarrow \boldsymbol{mea}_{has\text{-}intp}(s, m_1) \wedge \boldsymbol{mea}_{has\text{-}intd}(s, m_2) \wedge \boldsymbol{match}_{exact}(m_1, m_2).$$

[2] Throughout the rest of the paper, when we mention symbol s, we mean a symbol token - its occurrence in a field of a particular table tuple. So *'37.2°C'* is the occurrence of this symbol in row 1, column 2 of Table 1.

For example, $\boldsymbol{mea}_{\boldsymbol{match}}$('37.2°C', m_1, m_2) does not hold when m_1 and m_2 are temperatures of a patient measured at different time points. In general, we may want to know, for partially matched senses, how closely they match. For example, when two symbols s_1 and s_2 share their intended senses (e.g., because people recorded the same value with different precision), we can state the fact that "the interpreted sense of s_1 is closer than that of s_2 to their shared intended sense" as

$$\boldsymbol{mea}_{\boldsymbol{closer}}(s_1, s_2, m, m_1, m_2) \Leftrightarrow \boldsymbol{mea}_{\boldsymbol{has\text{-}intd}}(s_1, m) \wedge \boldsymbol{mea}_{\boldsymbol{has\text{-}intd}}(s_2, m) \wedge \boldsymbol{mea}_{\boldsymbol{has\text{-}intp}}(s_1, m_1) \wedge \boldsymbol{mea}_{\boldsymbol{has\text{-}intp}}(s_2, m_2) \wedge \boldsymbol{match}^{\boldsymbol{attr}}{}_{\boldsymbol{partial}}(m_1, m) \wedge \boldsymbol{match}^{\boldsymbol{attr}}{}_{\boldsymbol{partial}}(m_2, m) \wedge \boldsymbol{closer}^{\boldsymbol{attr}}(m, m_1, m_2).$$

5.3 The Purpose Aspect

This DQ aspect deals with the relationships involve the interpreted and expected senses of a symbol from the user perspective. As we have mentioned, an ultimate criterion for data quality is fitness for purpose. In our framework, the intended use of data values is captured through their expected senses. Therefore, quality issues arise when the interpreted and expected senses of a data value do not match exactly.

We are interested in a variety of relationships involving expected senses. Predicates such as $\boldsymbol{pur}_{\boldsymbol{match}}(s, m_1, m_2)$, for indicating the interpreted sense m_1 and expected sense m_2 of the symbol s match exactly, and $\boldsymbol{pur}_{\boldsymbol{closer}}(s_1, s_2, m, m_1, m_2)$, for indicating the interpreted sense m_1 of s_1 is closer than the interpreted sense m_2 of s_2 to their shared intended sense m, are defined in a similar way to their counterparts in the meaning aspect. The existence of expected sense, however, deserves more discussion.

Unlike the interpreted sense which is determined by the user directly, the expected sense is determined by a particular application. If a doctor is only interested in studying the effect of psychotherapy on the temperature of the patient, we'll say that the blood pressure (or more obviously the number of chairs in the room) have no expected senses to that doctor. To formalize this, let M_e denote a subset of M, determined by the tasks and goals the user has to fulfill. In our example, M might have temperatures and blood pressures taken at any time, while M_e might only have temperatures taken around noon. We say $m \in M_e$ is an expected sense of a symbol s if m matches, at least partially, with the interpreted sense of s. This can be stated as

$$\boldsymbol{pur}_{\boldsymbol{has\text{-}exp}}(s, m) \Leftrightarrow m \in M_e \wedge \exists m' \in M.\, \boldsymbol{mea}_{\boldsymbol{has\text{-}intp}}(s, m') \wedge (\boldsymbol{match}^{\boldsymbol{attr}}{}_{\boldsymbol{partial}}(m, m') \vee \boldsymbol{match}_{\boldsymbol{exact}}(m, m')).$$

This also allows us to consider the existence of a symbol, given partial knowledge about its expected sense. For example, we cannot find a symbol s in Table 1 with the property $\boldsymbol{pur}_{\boldsymbol{has\text{-}exp}}(s,$ *"Ben's cholesterol level on Nov. 5, 2007 at 13:05"*$)$.

When more than one expected sense exists, we may want to know if they are all comparable with respect to the interpreted sense of the symbol (so that later we can pick the closest one):

$$\boldsymbol{pur}_{\boldsymbol{comparable\text{-}exp}}(s) \Leftrightarrow \exists m \in M.\, \boldsymbol{mea}_{\boldsymbol{has\text{-}intp}}(s, m) \wedge \forall m_1, m_2 \in M_e.\ \boldsymbol{pur}_{\boldsymbol{has\text{-}exp}}(s, m_1) \wedge \boldsymbol{pur}_{\boldsymbol{has\text{-}exp}}(s, m_2) \rightarrow \boldsymbol{closer}^{\boldsymbol{attr}}(m, m_1, m_2) \vee \boldsymbol{closer}^{\boldsymbol{attr}}(m, m_2, m_1)$$

For example, given a temperature value *'37.2°C'* with its interpreted sense "the temperature quale of Ben measured at 13:05 with some thermometer", and two expected senses "temperature qualia of Ben measured at 13:05 with an oral/tympanal thermome-

ter", then these two expected senses are probably not comparable, unless we have a theory on how different types of thermometers affect temperature measurement.

5.4 The Trust Aspect

This DQ aspect deals with the relationships involve the intended and supposed senses of a symbol from the producer perspective. According to [20], in order to establish audience trust, both the sincerity and authority conditions have to hold. In the context of our framework, this means the user has to believe that the producer is neither a liar (i.e., no discrepancy caused intentionally, e.g., due to falsification) nor a fool (i.e., no discrepancy caused unintentionally, e.g., due to observation bias). Trust issues arise therefore when there is discrepancy between intended and supposed sense. Predicates in the aspects, such as $\boldsymbol{tru}_{\textit{has-sup}}$, $\boldsymbol{tru}_{\textit{comparable-sup}}$ and $\boldsymbol{tru}_{\textit{match}}$ are defined in the similar way as their counterparts in the purpose aspect. For lack of space, we do not elaborate them here.

6 Mapping Data Quality Attributes

We evaluate our approach by expressing quality attributes defined in the literature in our framework. One observation from this exercise will be that a single quality attribute often has multiple, sometimes conflicting, definitions. We differentiate these definitions by expressing them in terms of different (combinations of) theoretical quality predicates we have defined. This also allow us to accommodate competing views on how these attributes should be related, by making explicit the exact meaning of the attributes involved, and by distinguishing relationships that exist by definition and those that exist based on assumptions. Finally, this exercise also allows us to point out possibly new definitions.

6.1 Accuracy, Precision and Currency

Accuracy is normally understood as free of defects or correspondence to reality [24,13]. In [32], it is defined formally as the closeness between two representations s and s', where s' is the correct representation of the real-life phenomenon *s aims to* represent. If we accept that "correctness" here means "justified by some accepted standards or conventions", and make "closeness" be "identity" to get a Yes/No predicate, then this definition can be stated in terms of our symbol, meaning and trust aspects

$$\boldsymbol{accuracy}_{\textit{symbol}}(s) \Leftrightarrow \exists\, m \in M,\, s' \in S.\ \boldsymbol{mea}_{\textit{has-intd}}(s, m) \wedge\ \boldsymbol{tru}_{\textit{has-sup}}(s', m) \wedge \boldsymbol{sym}_{\textit{match}}(s, s').$$

According to this definition, we cannot have synonyms such as *'37.0°C'* and *'98.6°F'*, which may have been desired. To accommodate this, we can change the perspective from a fixed phenomenon to a fixed representation [25]; it defines accuracy as the closeness between two real-life phenomena m and m', where m is what a symbol s aims to represent and m' is what s appears to represent. This view requires only the meaning aspect

$$\boldsymbol{accuracy}_{\textit{meaning}}(s) \Leftrightarrow \exists m_1, m_2 \in M.\ \boldsymbol{mea}_{\textit{match}}(s, m_1, m_2).$$

The fact that s_1 is more accurate than s_2 can then be represented in this view as

$$\textbf{\textit{accuracy}}_{\textit{meaning-compare}}(s_1, s_2) \Leftrightarrow \exists m, m_1, m_2 \in M.\ \textbf{\textit{mea}}_{\textit{closer}}(s_1, s_2, m, m_1, m_2).$$

A typical understanding of precision as a quality attribute is the degree of details data values exhibit. For example, precision of numeric values is often measured by the number of significant digits used [5]. A number (e.g., *'3.1415926'*) is more precise than another one (e.g., *'3.14'*), assuming both represent the same phenomenon (e.g., the mathematical constant π), can be stated as

$$\textbf{\textit{precision}}_{\textit{symbol}}(s_1, s_2) \Leftrightarrow \textbf{\textit{sym}}_{\textit{more-detail}}(s_1, s_2) \wedge \exists m_1, m_2 \in M.\ \textbf{\textit{mea}}_{\textit{has-intd}}(s_1, m_1) \wedge \textbf{\textit{mea}}_{\textit{has-intd}}(s_2, m_2) \wedge\ \textbf{\textit{match}}_{\textit{exact}}(m_1, m_2)$$

Precision is often considered in close relation to accuracy. A typical intuition is that low precision leads to inaccuracy [25,5], which however cannot be accommodated by $\textbf{\textit{precision}}_{\textit{symbol}}$ alone. This is because having greater degree of details doesn't guarantee a better interpretation towards the intended meaning. In order to support this intuition, we need a strengthened notion of precision

$$\textbf{\textit{precision}}_{\textit{strengthened}}(s_1, s_2) \Leftrightarrow \textbf{\textit{precision}}_{\textit{symbol}}(s_1, s_2) \wedge \textbf{\textit{accuracy}}_{\textit{meaning-compare}}(s_1, s_2).$$

From the opposite view, one considers accuracy as a prerequisite for precision: in order to say s_1 is a more precise than s_1, both have to be accurate (i.e., have matching intended and interpreted senses). This view can be defined as

$$\textbf{\textit{precision}}_{\textit{meaning}}(s_1, s_2) \Leftrightarrow \textbf{\textit{sym}}_{\textit{more-detail}}(s_1, s_2) \wedge \exists m_{11}, m_{12}, m_{21}, m_{22} \in M.\ \textbf{\textit{mea}}_{\textit{match}}(s_1, m_{11}, m_{12}) \wedge \textbf{\textit{mea}}_{\textit{match}}(s_2, m_{21}, m_{22}) \wedge\ \textbf{\textit{match}}_{\textit{exact}}(m_{11}, m_{21}).$$

Now we really have a theorem $\textbf{\textit{precision}}_{\textit{meaning}}(s_1, s_2) \rightarrow \textbf{\textit{accuracy}}_{\textit{meaning}}(s_1) \wedge \textbf{\textit{accuracy}}_{\textit{meaning}}(s_2)$.

Currency as a DQ attribute is normally understood as the degree to which data are up to date [3,22]. As a first try, we could represent this understanding as:

$$\textbf{\textit{currency}}_{\textit{naive}}(s_1, s_2) \Leftrightarrow \exists m_1, m_2 \in M.\ \textbf{\textit{mea}}_{\textit{has-intd}}(s_1, m_1) \wedge \textbf{\textit{mea}}_{\textit{has-intd}}(s_2, m_2) \wedge\ \textbf{\textit{t}}(m_1) > \textbf{\textit{t}}(m_2)$$

where $\textbf{\textit{t}}$ returns the time component of a sense. One might notices that this definition allows us to compare the currency of the temperatures of different patients. When this is not desired, we can strengthen it using the notion of partial match

$$\textbf{\textit{currency}}_{\textit{strengthened}}(s_1, s_2) \Leftrightarrow \exists m_1, m_2 \in M.\ \textbf{\textit{mea}}_{\textit{has-intd}}(s_1, m_1) \wedge \textbf{\textit{mea}}_{\textit{has-intd}}(s_2, m_2) \wedge \textbf{\textit{match}}^{\textit{attr}}{}_{\textit{partial}}(m_1, m_2) \wedge\ \textbf{\textit{t}}(m_1) > \textbf{\textit{t}}(m_2)$$

Currency defined in this way is orthogonal to accuracy. As with precision, some authors consider a value s_1 is more current than another one s_2 only when both are accurate at a certain point in time [25]. This view can be captured by

$$\textbf{\textit{currency}}_{\textit{meaning}}(s_1, s_2) \Leftrightarrow \exists m_{11}, m_{12}, m_{21}, m_{22} \in M.\ \textbf{\textit{mea}}_{\textit{match}}(s_1, m_{11}, m_{12}) \wedge \textbf{\textit{mea}}_{\textit{match}}(s_2, m_{21}, m_{22}) \wedge \textbf{\textit{match}}^{\textit{attr}}{}_{\textit{partial}}(m_{11}, m_{21}) \wedge\ \textbf{\textit{t}}(m_{11}) > \textbf{\textit{t}}(m_{21})$$

A further complication, which will be discussed below, relates currency to relevance [5].

6.2 Relevance, Completeness and Timeliness

Relevance considers how data fits its intended use [13]. In its simplest form, it can be defined on the purpose aspect alone (recall M_e is a subset of M, determined by the tasks,

etc. the user of s has): $\textbf{\textit{relevance}}_{purpose}(s) \Leftrightarrow \exists m \in M_e.\ \textbf{\textit{pur}}_{has\text{-}exp}(s, m)$. This definition supports the view that relevance should evaluated before other quality attributes [5].

Intuitively, completeness concerns whether data is missing with respect to some reference set. In the simplest case, *value completene*ss [3,19] refers to the existence of null values in a reference column, row or table. This definition can therefore be understood as $\textbf{\textit{completeness}}_{symbol}(S_a) \Leftrightarrow \exists s \in S_a.\ \textbf{\textit{sym}}_{match}(s, \text{“null”})$, where S_a is the set of data values of interest. In a more complicated situation, *population completeness* [19] of S_a is defined as the existence of missing values with respect to the reference set M_e: $\textbf{\textit{completeness}}_{purpose}(S_a) \Leftrightarrow \forall m \in M_e\ \exists s \in S_a.\ \textbf{\textit{pur}}_{has\text{-}exp}(s, m)$. While the notion of completeness concerns whether every relevant data value is presented, we may also consider whether every presented value is relevant (the closest terms proposed in the literature for this attribute are “appropriate amount of data” [13] and “conciseness”[25]):

$$\textbf{\textit{completeness}}_{purpose\text{-}reverse}(S_a) \Leftrightarrow \forall s \in S_a\ \exists m \in M_e.\ \textbf{\textit{pur}}_{has\text{-}exp}(s, m).$$

When both conditions need to be enforced, we can define:

$$\textbf{\textit{completeness}}_{composite}(S_a) \Leftrightarrow \textbf{\textit{completeness}}_{purpose}(S_a) \wedge \textbf{\textit{completeness}}_{purpose\text{-}reverse}(S_a).$$

Some authors use timeliness to mean data is *sufficiently* up to date with respect to its intended use [3,21]. It can therefore be considered as another variant of currency [5]. The fact that a value s_1 is timelier than s_2 with respect to M_e can be stated as

$$\textbf{\textit{currency}}_{purpose}(s_1, s_2) \Leftrightarrow \textbf{\textit{currency}}_{meaning}(s_1, s_2) \wedge \textbf{\textit{relevance}}_{purpose}(s_1) \wedge \textbf{\textit{relevance}}_{purpose}(s_1).$$

6.3 Reliability and Believability

There is no generally accepted notion of reliability as a DQ attribute: some definitions overlap with that of accuracy [1], others are linked to dependability of the data producer [13], while still others are based on verifiability[16]. If we choose the last view -- that data is reliable if it can be verified (i.e., generated independently by different producers, possibly using different tools, methods, and etc.), we can define, given expect senses M_e

$$\textbf{\textit{reliability}}_{trust}(s) \Leftrightarrow \exists m_1 \in M,\ m_2 \in M_e.\ \textbf{\textit{tru}}_{match}(s, m_1, m_2).$$

This means what is intended to be represented by s matches exactly with what is supposed to be represented by it, according to the obligations the producer has. A violation of this condition may be caused by bias (i.e., lack of *objectivity* [3,24]) or intention (i.e., intentional *falsification* [13]) of the producer, or limitation of instrumentation, method, etc. Notice that reliability defined in this way is independent of accuracy. On the contrary, believability defined in [3,24] as “the extent to which data are accepted or regarded as true, real, and credible”, clearly concerns both the meaning and trust aspects

$$\textbf{\textit{believability}}_{meaning\text{-}trust}(s) \Leftrightarrow \textbf{\textit{accuracy}}_{meaning}(s) \wedge \textbf{\textit{reliability}}_{trust}(s).$$

7 Related Work

Some approaches to DQ share with ours the view that generic quality attributes (e.g., accuracy, completeness) may be understood in terms of more primitive quality constructs. In the Qurator project [15], such constructs (called *quality characterizations* or *QC*) are concrete, operational level quality attributes defined by scientists. For example, "accuracy" can be defined in terms of confidence *QC*, which can then be quantified using calculated number of experimental errors, or a function of the type of experimental equipment.

While the Qurator project provides a flexible way for specifying user-definable and domain-specific QCs in the context of e-Science, we are focusing on identifying primitive constructs that are reusable across domains. From a system-oriented view, [25] discusses various types of problematic correspondences (called of representation deficiencies) between a real world system (RW) and an information system (IS). For example, an incomplete representation means some RW phenomena are not (or cannot be) represented in IS, while an ambiguous representation means multiple RW phenomena have the same representation in IS. We also consider mismatches, but emphasize the role of producer and user, and mental representations (senses), abandoning the objectivist view of IS.

We are also not alone in considering DQ from a semiotics perspective. Thus, [21] proposes to understand and classify quality attributes in terms of syntactic (i.e., conformity to stored metadata), semantic (i.e., correspondence to external phenomena) and pragmatic (i.e., suitability for a given use) quality categories. Although these distinctions are embedded in our definitions of "senses" and "DQ aspects", they are only used in [21] to provide a conceptual framework to classify quality attributes. We also define quality attributes in terms of primitive constructs derived from these distinctions.

8 Conclusion

In this paper, we have proposed a novel, compositional framework for understanding and defining DQ attributes in a precise and comparable way, based on the notion of signs. We have also sketched a theory of senses for individual values in a relational table, based on its semantics expressed using some ontology. We have shown in our framework how multiple, sometimes conflicting, definitions of a DQ attribute could be differentiated, and how competing views on relating these attributes could be accommodated.

However, understanding DQ is just a means, not an end for us. Our ultimate goal in this quest is a methodology for "data quality by design". We have proposed a general goal-oriented quality design process for databases [10,11]. This process starts with application-specific goals where application data requirements are elicited and organized into an ordinary conceptual schema; then quality goals are modeled and operationalized to introduce new and modify existing data requirements in the initial schema. An important step during this process is to identify potential risks that may compromise quality of application data. The theory of senses provides exactly such machinery for a risk-based analysis. During schema design, one has to decide which components of the senses of application data values need to be modeled as schema

elements (according to user's goals and assumptions); such decisions eventually affect the quality of the application data. For example, Doctor Sudha is able to understand correctly the temperature value *'37.2°C'* with respect to "when" it was measured, exactly because there is a "time" attribute in the *Patient* schema. However, the design decision to leave out other components (such as how it was measured, with what type of thermometer and by whom) contributes to Sudha's partially incorrect understanding of *'37.2°C'*. Our immediate next step is to refine the notion of senses and formalize partial match between senses, and use them to derive patterns of risk factors for database design.

References

1. Agmon, N., Ahituv, N.: Assessing Data Reliability in an Information Systems. Journal of Management Information Systems 4(2), 34–44 (1987)
2. An, Y., Borgida, A., Mylopoulos, J.: Discovering the Semantics of Relational Tables through Mappings. In: Spaccapietra, S. (ed.) Journal on Data Semantics VII. LNCS, vol. 4244, pp. 1–32. Springer, Heidelberg (2006)
3. Batini, C., Scannapieco, M.: Data Quality: Concepts, Methodologies and Techniques. Springer, Heidelberg (2006)
4. Bovee, M.: A Conceptual Framework and Belief-Function Approach to Assessing Overall Information Quality International. Journal of Intelligent Systems 18(1), 51–74 (2003)
5. Gackowski, Z.J.: Logical interdependence of data/information quality dimensions - A purpose focused view on IQ. In: Proc. of the 2004 International Conference on Information Quality (2004)
6. Calvanese, D., Giacomo, G.D., Lenzerini, M., Nardi, D., Rosati, R.: Data Integration in Data Warehousing. Journal of Cooperative Information Systems 10(3), 237–271 (2001)
7. Fitting, M.: Intensional Logic. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy (Spring 2007), `http://plato.stanford.edu/archives/spr2007/entries/logic-intensional/`
8. Grice, H.P.: Meaning. The Philosophical Review 66, 377–388 (1957)
9. Jeusfeld, M.A., Quix, C., Jarke, M.: Design and analysis of quality information for daa warehouses. In: Ling, T.-W., Ram, S., Li Lee, M. (eds.) ER 1998. LNCS, vol. 1507, pp. 349–362. Springer, Heidelberg (1998)
10. Jiang, L., Borgida, A., Topaloglou, T., Mylopoulos, J.: Data Quality by Design: A Goal-Oriented Approach. In: Proc. of the 12th International Conference on Information Quality (2007)
11. Jiang, L., Topaloglou, T., Borgida, A., Mylopoulos, J.: Goal-Oriented Conceptual Database Design. In: Proc. of the 15h IEEE Int. Requirements Engineering Conference, pp. 195–204 (2007)
12. Liu, K.: Semiotics in Information Systems Engineering. Cambridge University Press, Cambridge (2000)
13. Liu, L., Chi, L.N.: Evolutional Data Quality: A Theory-Specific View. In: Proc. of the 2002 International Conference on Information Quality (2002)
14. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A., Schneider, L.: WonderWeb Deliverable D17 (2002)
15. Missier, P., Preece, A.D., Embury, S.M., Jin, B., Greenwood, M., Stead, D., Brown, A.: Managing Information Quality in e-Science: A Case Study in Proteomics. In: ER 2005 Workshops, pp. 423–432 (2005)

16. Naumann, F.: Do metadata models meet IQ requirements? In: Proc. of the 1999 International Conference on Information Quality, Cambridge, MA, pp. 99–114 (1999)
17. Peirce, C.S.: Collected Papers. In: Peirce, C.S., Hartshorne, C., Weiss, P., Burks, A. (eds.), vol. 8. Harvard University Press, Cambridge (1931–1958)
18. Pernici, B., Scannapieco, M.: Data Quality in Web Information Systems. In: Proc of the 21st int. Conference on Conceptual Modeling, pp. 397–413. Springer, London (2002)
19. Pipino, L.L., Lee, Y.W., Wang, R.: Data quality assessment. Comm. of ACM 45(4), 211–218 (2002)
20. Price, G.: On the communication of measurement results. Measurement 29, 293–305 (2001)
21. Price, R., Shanks, G.: A Semiotic Information Quality Framework. In: Proc. IFIP International Conference on Decision Support Systems, Prato (2004)
22. Redman, T.C.: Data Quality for the Information Age. Artech House, Boston (1996)
23. Wang, R.Y., Reddy, M.P., Kon, H.B.: Toward quality data: an attribute-based approach. Decision. Support Systems 13(3–4), 349–372 (1995)
24. Wang, R.Y., Strong, D.M.: Beyond accuracy: what data quality means to data consumers. Journal of Management Information Systems 12(4), 5–33 (1996)
25. Wand, Y., Wang, R.Y.: Anchoring data quality dimensions in ontological foundations. Communications of ACM 39(11), 86–95 (1996)