Good Answers from Bad Data: a Data Management Strategy

Henry B. Kon Stuart E. Madnick Michael D. Siegel

December, 1995 Sloan WP# 3868

Good Answers from Bad Data: a Data Management Strategy

Henry B. Kon Stuart E. Madnick Michael D. Siegel

{hkon, smadnick, msiegel}@mit.edu
MIT Sloan School of Management
Informatio Technologies Group / E53-320
Cambridge, MA 02139

December, 1995

Sloan School of Management WP# 3868

ABSTRACT

Data error is an obstacle to effective application, integration, and sharing of data. Error reduction, although desirable, is not always necessary or feasible. Error measurement is a natural alternative. In this paper, we outline an *error propagation calculus* which models the propagation of an *error representation* through queries. A closed set of three *error types* is defined: attribute value *inaccuracy* (and nulls), object *mismembership* in a class, and class *incompleteness*. Error measures are probability distributions over these error types. Given measures of error in query inputs, the calculus both computes and "explains" error in query outputs, so that users and administrators better understand data error. Error propagation is non-trivial as error may be amplified or diminished through query partitions and aggregations. As a theoretical foundation, this work suggests managing error in practice by instituting measurement of persistent tables and extending database output to include a quantitative error term, akin to the confidence interval of a statistical estimate. Two theorems assert the completeness of our error representation.

KEYWORDS AND PHRASES

Data quality management, relational database, query processing, data error

ACKNOWLEDGEMENTS

This work was supported in part by ARPA and USAF/Rome Laboratories, under contract F30602-93-C-0160

1. INTRODUCTION

Bad data leads to bad decisions. Non-trivial error levels have been observed, however, in a variety of applications including receivables, inventory, financial, military personnel, and criminal justice [13, 16, 19]. Advances in data warehouses, data sharing networks, database mining, and on-line data analysis suggest that error in data should be properly managed - particularly on behalf of new users who may least understand imperfections in data.

U.S. government agencies such as the Census Bureau and Energy Information Agency use econometric models of under-count and other biases in survey results [3, 14, 17]. As shown in Table 1 below, statisticians have rich error models, while no quantitative error model is defined for the database area. Contrasting their data and operations helps to explain why error is dissimilar for the two disciplines.

	Data	Operators	Error
Statistics	aggregate quantitative	regression ⇒ estimates	sample bias, measurement error, confidence interval ⇒ rich models of error and its propagation
Database	object relationships instance-level detail	logical, set-based ⇒ deterministic	vague terms such as 'accuracy' and 'completeness' ⇒ no accepted model of error or its propagation

Table 1: Views of error across the statistics and database disciplines

Two components to error management strategies are *error reduction* and *error measurement*. Error reduction involves correcting existing data or improving a data collection process. Error measurement, the goal underlying our effort, involves assessment and reporting of error so users can be aware of error in data they use. Error reduction is often unnecessary and sometimes unfeasible. First, it presumes the control or resources necessary to improve the data. While not unlikely for locally administered data, this is inconsistent with norms for data sharing networks - where data are independently managed, of varying quality, and often unfamiliar. Second, not all application contexts require error-free data. An economically optimal state in data processing will involve some degree of error.

In this paper we begin a foundation for the measure and propagation of error in databases. An error representation is a description of error as to three error types: attribute value inaccuracy (and nulls), class incompleteness, and class mismembership. This set of three is closed under a probabilistic error propagation calculus which computes error in query output as a function of error in query inputs. 'Closed' here means that a table, derived from a table with these errors, will itself have only these errors.

Error measurement methods are not a main concern in this paper. But because our model depends on probability distributions of error as input - a natural question is: *Where do I get such distributions?* Probability distributions of error result from error measurement, and may correspond to beliefs, statistics, theory, etc. concerning both direct and indirect evidence of error such as "timeliness", "source", or "collection method". Statisticians [3] and market researchers [9, 10] have techniques for estimating error in surveys. These include administrative record matching (comparing data to reference sources of related information) and post-enumeration surveys (more rigorous follow-up surveys on a portion of the population). We are currently designing a knowledge acquisition system for elicitation from DBA's - information leading to error distributions, including both formal and intuitionistic evidence.

We recognize error in any of the following, among others:

- updates to old data,
- fixes to known untruths in data,
- heterogeneous definitions,
- discrepancies among two "redundant" data sources,
- differences between opinions, forecasting, or simulation models.

The salient aspect in each of these is that there are two conflicting accounts, i.e., logical interpretations [5] of the "same" world (e.g., database schema or context [7]). Given interpretations r and d of the world, where r is a *reference source* for a data set d, a description of error in d constitutes the full set of corrections to d as implied by r. We are modeling error in this sense as a "difference" between tFwo data sets. This is analogous to treating scalar error as a difference between two numbers [2].

In practice, we envision a software system, based on this theory, which computes an error measure for each data set derived from a database. (A DBA or software agent would produce a priori some error measure on the input relations.) The system would take as input an error term which is a set of error measures on persistent data[1]. This approach provides users a sense of output data reliability, and it amortizes the cost of measurement over various queries run. This approach also informs the DBA as to which data quality improvements will yield the best return for users. We believe that this technology has application to all data, whether in a stand-alone database, a data warehouse, or shared in a multi-database network. This research will lead to a set of Data Quality functions, shown below in the data

[1] "Atomic" data which are stored and maintained over time.

warehouse setting as an example. The "confidence interval" here is a set of probability distributions about error. Error propagation calculus (EPC) execution is "induced" by query operations.

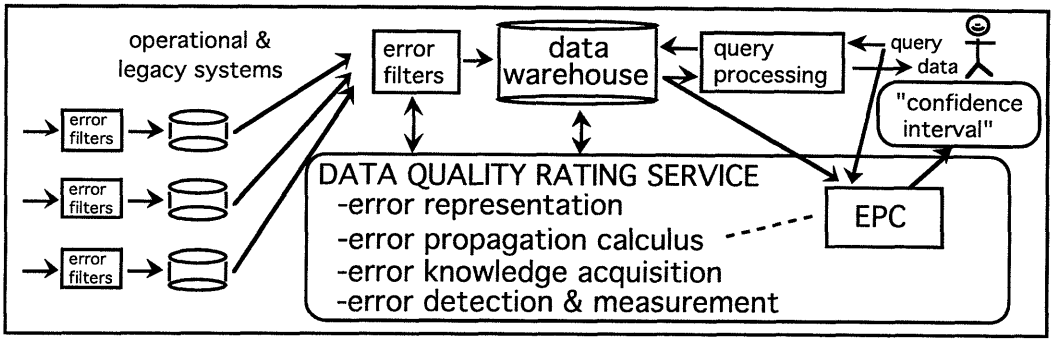

Figure 1: A data quality subsystem in an on line analysis setting

Error and its propagation are considered in survey methods [6], numerical analysis [2], and applied statistics [10]. As was shown in Table 1, this stream of work is concerned with error in quantitative data (e.g., numbers and counts) and with the propagation of error through regressions and basic arithmetic. These operations are different from those in a database query, however. Another stream of research on database incompleteness and uncertainty concerns querying over null values and over disjunctive and probabilistic representations of non-determinism [4, 8, 12, 25]. Our focus is on quantitative modeling of error, and not on extending the database to represent non-determinism. (Error and uncertainty are orthogonal[2] in this sense.) Motro [20] considers completeness and validity of data, but defined at the "tuple level" and with no quantitative interpretation of error. In [22], a similar model to ours is described, but formal properties are not analyzed, incompleteness is not covered[3], and assumptions are made about uniformity and independence in probabilities.

In Section 2 we describe a logical error representation. In Section 3 we present the probabilistic calculus and representation. In Section 4 we conclude, considering possibilities for application of these ideas in practice.

2. LOGICAL REPRESENTATION FOR DATABASE ERROR

Error-related concepts such as accuracy, completeness, integrity, and consistency are often vague. They apply differently to a data set than to a datum, and they may be operationalized differently for categorical and numeric data. A precise, general, and quantitative formulation of error would be useful.

---

[2] We may have probabilities but no error, e.g., P(coin toss = HEAD) = .5. Or we may have error but no probabilities if we "know error deterministically" (e.g., Joe earns $40K not $30K). Uncertainty about error is typical, however.

[3] e.g., an empty relation by definition contains no errors

Example 2.1 (the three error types) Table 2 below lists alumni of a college who have made large donations and live in the U.S. It includes their Name, State, Degree, Major, Year, and Donations.[4] Three error types are described by example beneath Table 2 and will be formalized in Section 2.2.

2.1 Examples of error and its propagation

Alumni (U.S. active alumni)						
Record #	**Name**	**State**	**Degree**	**Major**	**Year**	**Donations**
1	Mae Ray	RI	BS	Law		$ 800,000
2	Lisa Jordan	RI	MS	CS	1982	**$ 500,000**
3	Tom Jones	ME	MS		1988	$ 420,233
4	Jon Wales	CT	BS	Eng'g	1953	$ 600,000
5	Frank Ames	NY	PhD	Biology	1986	$ 820,000
6		VT	MS	Business		$ 243,000

Table 2: Example table with errors - Alumni relation

inaccurate attribute value (record #2): Lisa Jordan's donation amount is actually **400,000**. Included in this category is the ***null attribute value*** (record #1&3): These are values which exist but are unknown to the database. Mae Ray's Year of Graduation is 1991. Tom Jones' major was CS.
class incompleteness: A tuple for the alumnus John Junto is incomplete, i.e., missing. (Perhaps the corrupted record #6 was his). He is an alumnus who satisfies the class definition, but is not listed.
class mismembership (record #4&6): Jon Wales is a mismember - he is no longer active. Record #6 is a mismember as it represents a spurious object - null or corrupted keys are modeled as mismembers.

◆ (end example)

These types of errors may be found wherever data collection can not be tightly controlled, or where error is not a major concern. We usually understand error not as corrections to particular facts (i.e., logically as in Example 2.1), but only as probabilities as in Example 2.2 below.

Example 2.2 (error as probabilities) A DBA for the Alumni database we have worked with believes that approximately three hundred records in his data incorrectly list an alumnus as Deceased. He suspects that this is due both to mis-keying of ID's in data entry, and to falsification by alumni (to prevent solicitations!). In this case, error can only be modeled as probabilities of error for various subsets of the population. ◆

Example 2.3 below provides an intuition into error propagation.

Example 2.3 (propagation) Table 3 below results from the following select-project query on Alumni Table 2 and errors of Example 2.1. *Select Name, Donations From Alumni Where Degree = 'MS'.* Three errors from the input are propagated

[4] This is an illustration of a 100,000-record university database we work with in our research. Data are hypothetical.

to the output. First is the inaccurate value 500,000. Second is the blank-key mismember. Third is incomplete John Junto whose degree is in fact 'MS'. ◆

Alumni-donation	
Name	**Donations**
Lisa Jordan	**500,000**
Tom Jones	420,233
	243,000

Table 3: Result of standard select-project query on Alumni relation

2.2 Logical Error Representation

2.2.1 A conceptual model of error

We define error in a database table relative to a *state* (e.g., "true" state) of the world as in Figures 2 and 3 below. The state of the world concerns objects, their attribute values, and their membership in classes. Let *true data* (D_t) be that unique, error-free, and *possibly unknown* table that correctly represents the state of the world (in the context of the table's scheme D). The data in the database table (D_a) is an approximation of that world. Data error (D_e) is a representation of *difference* between D_t and D_a. We will define a correction operation $\oplus$ such that $D_t = D_a \oplus D_e$ in Section 2.2.3. We write $\oplus r$ and $\oplus t$ for true r relation and true t tuple respectively.

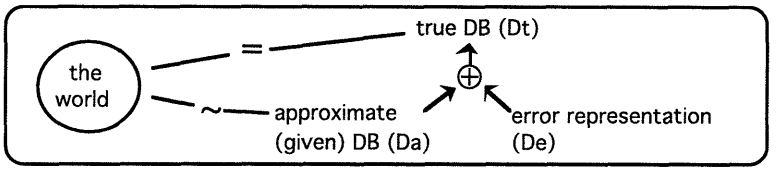

Figure 2: Data error: Dt = Da ⊕ De

We construct our error representation over a semantics having *class statements* and *attribute statements*. A class statement assigns an object to a category. An attribute statement associates a value with an attribute of an object. Figure 3 below summarizes our concept of error and its relation to these meanings.

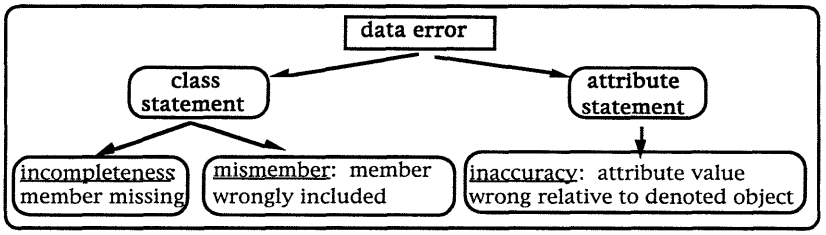

Figure 3: The three error types and their relation to meaning of data

For class statements, key values denoting class members should denote ALL members (else incompleteness) and ONLY members (else mismembership). For an attribute statement, error corresponds to inaccurate (e.g., untrue) and blank (null) attribute value vis-a-vis an object. These error types will be shown to fully characterize the difference between any relation and any corresponding true (i.e., fully correct) relation. Our scheme will be operationalized in this paper for the relational data model[5].

2.2.2 Error and the relational model

Definition 2.1 (error-free relational table) A *relation* r on scheme R is a subset of the cartesian product of R's domains. We interpret every table as representing a *class*. Key values represent objects in the class, and attribute columns contain their attribute values. Key values may span multiple columns. A *tuple* t is an element of r where $t.j$ denotes the sub-tuple of t on column(s) j. R's class is represented by the key of R. Non-key columns are *attributes* of objects denoted by the key. Keys in persistent relations are assumed to be unique. Initial model inputs are assumed to be 3NF. ◆

Definition 2.2 (error definitions) Let K be the key for table scheme D. Then $D_t.K$ is the true set of objects in D's class. Let C_i be the set of objects *incomplete* from D_a, so $C_i = D_t.K - D_a.K$. Let C_m be the set of objects *mismember* in D_a, so $C_m = D_a.K - D_t.K$. Let $t.K$ be the key value of a tuple t on D. We say that an attribute value $t.j$ ($j \not\subset K$) is *inaccurate* if $t.j \neq \oplus t.j$. A null ' ' or syntax error in an attribute value means "value not known" (in databases and for us: ' ' $\neq c$ for all constants c). A null or syntax error in a key is modeled as a mismember because "the object" is spurious or not identified. ◆

Definition 2.3 (error representation) If r is a table then r_{em}, r_{ea}, and r_{ei} are its *error tables* (Table 4 below). These tables have the same scheme as r and contain *error tuples* to indicate respectively: mismembers, inaccuracies, and incompleteness vis-a-vis r. The *error triple* $r_e = \langle r_{em}, r_{ea}, r_{ei} \rangle$ constitutes our *logical error representation*, which is fundamentally a correction to r. We say r exists in an *error state*, denoted by r_e, which maps r to the table $\oplus r$ representing the state of the world. ◆

Example 2.4 (error triple r_e) The error triple $\langle r_{em}, r_{ea}, r_{ei} \rangle$ for table and errors of Example 2.1 is in Table 4 below. Tuples in r_{em} and r_{ei} denote mismember and incomplete tuples of r respectively. Tuples in r_{ea} represent member tuples of r having one or more inaccurate attributes. Values in the attribute columns of r_{ea} are accurate and are only present (i.e., not '–') when the corresponding value in r is inaccurate (tuples 1-3 below). All attribute column values in r_{ei} are present and

[5] our system is logical, however, so that the data model chosen should be incidental

accurate (tuple 6 below). In r_{em} only the key column values are of concern and not attribute column values. The logical representation is an exact account of error in r relative to a state of the world. ◆

r_{em} Alumni **mismember**						
4	Jon Wales					
6	@ (NULL)					

r_{ea} Alumni **inaccurate**						
1	Mae Ray	_	_	_	1991	_
2	Lisa Jordan	_	_	_	_	400,000
3	Tom Jones	_	_	CS	_	_

r_{ei} Alumni **incomplete**						
6	John Junto	VT	MS	Psychology	1984	243,000

Table 4: Sample error triple: $<r_{em}, r_{ea}, r_{ei}>$

To summarize the logical representation - a tuple in r is either: (1) fully correct (member with no inaccuracy), (2) mismember , or (3) inaccurate (member with inaccuracy). An object in the world may be incomplete from r and would then be in r_{ei}. These define a partition on $r \cup r_{ei}$, the space of relevant tuples. This "deterministic" concept of error will serve as a mathematical basis for our probabilistic scheme which, unlike the logical model, admits measurement uncertainty.

2.2.3 The correction operator

The correction operator $\oplus$ has two forms: a tuple correction $\oplus_t(r_1, r_e)$ for $r_1 \in r$, and a relation correction $\oplus_r(r, r_e)$. A relation is corrected by eliminating mismember tuples, inserting incomplete tuples, and correcting inaccurate tuples: $\oplus_r(r, r_e) = \{\oplus_t(r_1, r_e) \mid \exists r_1 \in r \wedge r_1.K \notin r_{em}.K\} \cup r_{ei}$. An inaccurate tuple is corrected by replacing inaccurate and null attribute values with accurate values. These are written in the calculus as $\oplus r$ and $\oplus r_1$. (For brevity, the formalism for tuple correction is not given.)

We now show a simple yet important completeness property of our error triple and correction operator. Let D be the set of possible instances of a relation r (including possible errors). Let W be the set of possible instances of r corresponding to a state of the world (e.g., no nulls, spurious objects, syntax errors) so that $W \subset D$. The following theorem states that this scheme can characterize the discrepancy between any $<d, w>$ pair. E is the space of possible error triples under the associated relation scheme.

Theorem 2.1 (logical representation completeness) The correction operator $\oplus_r$ and error triple define an onto mapping from D to W (i.e., can map any $d \in D$ to any $w \in W$) such that $W = \oplus_r(D, E)$.

Proof (sketch) An object determines its true attribute values. The true relation is therefore determined by the true set of objects in its class. By definition, an error triple defines which objects are truly in a class and which are present but do not belong, as well as defining their true attribute values. Assuming sufficient domains, an error triple can thus map any $d \in D$ to any $w \in W$. ◆

This completes the description of the logical representation. Whereas data quality is sometimes discussed vaguely as "accuracy and completeness", we have formalized error using a correction mapping over a finest-grain discrete space. This will become the sample space[6] of the probabilistic error representation. In our model, error takes two forms: class-level (incompleteness and mismembership) and attribute-level (inaccuracy and nulls). This is consistent with the literature on statistical methodology, in which measurement error is considered at the population level and the individual level (sampling and non-sampling error) [6]. In [15], we show (and prove correctness for) a logical error propagation calculus which propagates this representation through select, project, cartesian product, natural join, and aggregations - in a closed and lossless fashion. This covers the space of conjunctive queries [1], the basis for SQL. We continue with the probabilistic model next.

3. PROBABILISTIC ERROR REPRESENTATION & PROPAGATION CALCULUS

Our logical model represents error as deterministic corrections to individual facts. Inaccuracy in a number would be denoted by a number (e.g., the true value, numeric difference, or percentage deviation). In practice, error may only be estimated by subjective judgment, actuarial method, etc. Therefore, our probability model represents error as distributions. Inaccuracy in a number may then be represented as a distribution of possible numbers. Examples 3.1 and 3.2 below illustrate our probabilistic representation and the conditional probability distributions of error that will be used.

Example 3.1 (probabilistic inaccuracy representation) The Income_table below represents taxpayers with an identifier (SSN), declared income (I), and employment category (EC).

[6] mutually exclusive, collectively exhaustive, and finest-grain enumeration of events of interest [11]

Income_table		
SSN	**Income (I)**	**Employment Category (EC)**
142-06-4480	46,514	self-employed
113-22-4231	21,245	retired
775-95-4573	29,223	corporate
...	...	...

Let E be the inaccuracy in an income figure (i.e., the reported value minus true value). We consider three cases (below). In (i), E is unconditional: income under-reporting is independent of other attributes. In (ii) E is conditioned on I: under-reporting varies with income. In (iii) E is conditioned on I and EC: the self-employed under-report more so than retirees or corporate employees.

(i) E independent of I & EC:	$P(E=e)$	$= f(e)$
(ii) E depends on I:	$P(E=e \mid \oplus I=i)$	$= f(e, i)$
(iii) E depends on I and EC:	$P(E=e \mid \oplus I=i \wedge \oplus EC=ec)$	$= f(e, i, ec)$ ◆

Practically speaking, we believe that an error representation should be able to provide the following kinds of information: (a) the probability that a given (e.g., random) tuple is mismember, (b) the probability that a given tuple is inaccurate and that its true value is x (a tuple variable), (c) the probability that a given class member is incomplete, and (d) the probability that a member has value x given that it is incomplete. Information such as this - when properly structured - will allow us to understand a variety of biases that may be "hidden" in the data. Our representation will cover these and these lead naturally to a calculus for aggregate queries such as count and sum [15].

In Example 3.1 we showed conditional distributions representing inaccuracy. The use of conditionals for mismembership is illustrated by Example 3.2 below.

Example 3.2 (conditional mismembership representation) Consider a snapshot of a Student table. Assume the snapshot not been refreshed (updated) in one year. The probability of a tuple being a mismember (e.g., the student in reality having graduated) will be conditional upon the Year the student entered and on the Degree (PhD, MS, BS) being pursued. ◆

The probability formalism is as follows. We model the measurement of error in a table r as an experiment. The error state space of the logical representation $D \times W$ is the sample space over which probabilistic representation and calculus are defined. Because w is a function of $<r, r_e>$ (Theorem 2.1), then a probability distribution p_1 on the set of possible error states (e.g., error triples r_e) of r - defines a probability distribution on the true state of the world. Given r and p_1, we therefore have "perfect probabilistic information" about the world. This is an important feature for any

model of error. We will state this as Theorem 3.1. Probabilistic statements about mismembership, inaccuracy, and incompleteness are one way of encoding knowledge leading to probability distributions of error.

Query aggregations and partitions amplify some errors while diminishing others. Even for analysts who understand error in persistent data, it may be unclear how the error impacts data output. And outputs, not inputs, are the concern of users. We now begin with the probabilistic representation and calculus. We describe a full select calculus, and consider projection by example. For brevity, some of the calculus sub-expressions are presented but not explained. This is hopefully sufficient to illustrate the approach.

3.1 Probabilistic Select Calculus

Each calculus equation computes an output event probability from input event probabilities. Each therefore describes an error propagation mechanism, or "dependency structure" by which error migrates from input to output. Let tables r and s be input to and output from a selection. The scheme of both is R, with r_e and s_e as error triples. Let $K \subseteq R$ be the key of R. As an example of the probabilistic events we will consider, let s_1 be a random tuple drawn from the result s. Let $s_1.K \in s_{em}.K$ be the event that $s_1.K$ is a mismember of s. Then $P(s_1.K \in s_{em}.K)$ is the probability of this event. Sometimes, we wish to assign a higher likelihood or degree of error to some subsets of the data(world) than to others, as in examples 3.1 and 3.2 above. A conditional distribution such as $P(s_1.K \in s_{em}.K \mid s_1 = x)$ allows us to do this (where x is a tuple variable on R). (Of course $P(s_1.K \in s_{em}.K \mid s_1 = x) = P(s_1.K \in s_{em}.K)$ if independence is assumed.)

We start with select mismembership. A selection condition is a logical formula f [24]. If a tuple t satisfies f we write $f(t)$ else $\neg f(t)$.

mismembership in the select result:

$$P(s_1.K \in s_{em}.K \mid s_1 = x) = P(s_1.K \in r_{em}.K \mid s_1 = x) \quad (1a)$$
$$+ P(s_1.K \in r_{ea}.K \wedge f(s_1) \wedge \neg f(\oplus s_1) \mid s_1 = x) \quad (1b)$$

Logically, error propagation calculus (sub-)expressions 1a and 1b above state that two exclusive events among input tuples can result in an s mismember. 1a covers the event that $s_1.K$ was a mismember in r, in which case (by definition of a selection) it is also a mismember in s. **Example of 1a**: *Select * from Alumni where State = "CT"* yields Jon Wales as a mismember of s as he was a mismember of r. 1b describes the other way a tuple may be a mismember in s - when an inaccuracy in r causes a tuple to be wrongly selected into s. **Example of 1b**: *Select * from Alumni where Donations*

> *450,000* yields Lisa Jordan as a mismember in *s*. Her actual donation was 400,000 while it was listed as 500,000 in the database. Although her database tuple satisfies f, her "true tuple" does not.

The probability of an output mismembership is a function of the probabilities of the these two types of input error events. The probability that a random output tuple $s_1 \in s$ is a mismember (given $s_1 = x$) is the probability that, for the tuple $s1 \in r$, and given $f(s_1)$, then what is the conditional probability - in r - that s_1 is either a mismember of r or s_1 is inaccurate resulting in false selection by f. And, because of the conditionals, a probabilistic "filtering" of error is going on. The selectivity of f over conditioning variables may lead to different proportions of tuples in each error category for r and s. The propagation of probability distributions is based on the underlying logical structure defined in [15]. Using a commutativity-based proof method, we show that this logical representation of error can be propagated very naturally under relational selection, projection, cartesian product, natural join, and aggregation.

The select calculus expression for inaccuracy is covered next. Whereas mismembership concerns objects and classes, inaccuracy concerns an attribute value vis-a-vis an object. As in mismembership, we adopt a conditional interpretation of inaccuracy. y below is another tuple variable on R.

inaccuracy in the select result:

$$\boxed{P(s_1.K \in s_{ea}.K \wedge \oplus s_1 = y \mid s_1 = x) = P(s_1.K \in r_{ea}.K \wedge \oplus s_1 = y, f(\oplus s_1) \mid s_1 = x)} \quad (2a)$$

This equation describes the single event in the input event space that results in an inaccuracy in the output. This is the case where an inaccurate tuple s_1 of r satisfies f. $f(\oplus s_1)$ ensures that $s_1.K \notin s_{em}.K$. **Select inaccuracy example of 2a**: *Select * from Alumni where donation > 300,000* results in Lisa Jordan as an inaccurate tuple in *s*. She is not a mismember of *s* as her donations are in fact greater than 300,000.

To conclude the select calculus we consider incompleteness. Let *o* be a random tuple from $\oplus s$. Let t be the corresponding inaccurate tuple[7] in r such that $t.K = o.K$. The reasoning behind calculus expression 3a-b below is analogous to that of the mismembership expression 1a-b. P_s and P_r represent probabilities on s and r respectively.

[7] if its existed, i.e., if object *o* was not incomplete

incompleteness[8] in the select result:

$$P(o.K \in s_{ei}.K \mid o = x) = P(o.K \in r_{ei}.K \wedge f(o) \mid o = x) \quad (3a)$$

$$+ P(o.K \in r_{ea}.K \wedge f(o) \wedge \neg f(t) \mid o = x) \quad (3b)$$

$$P_S(o = x) = P_r(o = x \mid f(o)) \quad (3c)$$

3.2 Probabilistic Project Calculus

Example 3.3 (project propagation) The projection *Select Major From Alumni* (Table 1) results in the table below. This is the set of majors among active US alumni - the class of alumni represented by the input relation. Notice that there are no attributes (only keys) in Major. Business, which occurred as an attribute value of a mismember in Alumni, is a mismember in Major since it was not the correct Major for any active US alumnus. Psychology, the major of incomplete alumnus John Junto is incomplete. (It would not have been if it were the listed value for any other tuple in Alumni). ◆

Major	
Law	
CS	
Eng'g	
Biology	
Business	

('Psychology' is incomplete)

('Business' is mismember)

In the example above, Major was an attribute column in context of Alumni, but "became" an object in the Major table due to the projection. A propagation calculus must account for such semantic transformations and convert across error measures from one interpretation of data to another (e.g., as in 1b & 3b).

Let r and s be input and output respectively for a projection: $s = \pi_S(r)$. Probabilistic project propagation depends on the relationship between the projection list S and the key K of the input scheme R. If S includes the entire key of R, then the key of S and R are the same, and the incompleteness and mismembership of s and r are the same. If the key is removed, then a new key arises (as in Major above) and error must be computed accordingly. Another factor in the projection is the relationship between S and the set of columns that are conditioning in the error representation. If conditioning columns are removed, then a new marginal distribution of error must be computed for on remaining columns in order to maintain error information. Example 3.4 below describes the calculus for projection incompleteness when the conditioning columns are kept and the key is removed.

[8] An alternative formulation of incompleteness is not shown for brevity.

Example 3.4 (project incompletes - conditioning columns kept, key removed) The query and resulting table of Example 3.3 illustrate this case. Let R_k be the key of R. Because S is disjoint from R_k, there is a key change so that $S_k = S$.[9] Define o as in 3a-c above. We will compute incompleteness as

$P(o.K \in s_{ei} \mid o.K = x)$. $P(o.K \in s_{ei} \mid o.K = x) = P(x \in s_{ei}) = P(x \in \oplus r.S \wedge x \notin r.S)$.

$P(x \in \oplus r.S \wedge x \notin r.S) =$ *(a)* 0 for $x \in r.S$, and *(b)* for $x \notin r.S$: $P(x \in r_{ei}.S \vee x \in \{\oplus t \mid \oplus t \in r_{ea}.K\}) = 1 - P(x \notin r_{ei}.s \wedge x \notin \{\oplus t \mid t \in r_{ea}.K\})$. ◆

This error propagation calculus expression indicates that object $o.S_k$ will be incomplete from s if either incompleteness or inaccuracy in r masked the fact that a member of r in fact had S value $o.S_k$.

This completes our presentation the error representation and propagation calculus formalism. As seen from this model, error propagation is non-trivial but computable (discussed more formally in [15]).

3.3 Probabilistic representation and a property

"Knowing error fully" in a probabilistic sense implies having a probability distribution on the error sample space, so that each error state is individually quantified for likelihood (i.e., so that there is no ambiguity in error probabilities). A probabilistic error representation - consisting of expressions such as those in the left hand sides of calculus expressions 1, 2, and 3 - is one way of defining these probability distributions. (The right hand sides tell us how these errors propagate and interact from input to output.)

We present below a probabilistic analog to the logical Theorem 2.1. It states that - given a table d - and given a probability distribution over E (the set of possible error triples under d's scheme), then a unique probability distribution on W is specified. Therefore, a table and its error distribution provide a unique and unambiguous probability distribution on the true state of the world. This is an indicator of the model's fidelity to an underlying objective concept of truth, which is a probability distribution on possible states of the world, given all available information.

Theorem 3.1 (probabilistic representation completeness) A table d, and the probability distribution on E together define a probability distribution on W.

Proof: Let w be any point in W. We know from Theorem 2.1 that there exists one or more points in $<D, E>$ which map to w, i.e., are in a state such that $\oplus_r(d, e) = w$. These are the pre-image of w in $<D, E>$. Here, D is fixed at d, so the probability of w is simply the union of the probabilities of all points in E which

[9] We assume no transitive dependencies in R as projection input is 3NF under standard optimization heuristics [24].

are in the pre-image of w, which is given by the distribution on E. Therefore each element of w is individually quantified for error. ◆

Many uncertainty models embody assumptions about the "shape" of uncertainty (e.g., uniformity, independence, and normality [4, 8, 25]). These may or may not be valid in a given setting. The current model makes no assumptions about distributions, but only specifies what probabilities are relevant.

4. DISCUSSION

We have formulated an original model of the structure of error and described its propagation in databases. We began with a conceptual model, described two error representations, and outlined a calculus for their propagation through queries. The logical model defines a semantics and the probabilistic model provides a quantitative interpretation. Error is defined as a difference between two relations, based on a mathematically defined point space. The error propagation calculus is a set of operators which compute output error as a function of input error. The set of error types - incompleteness, mismembership, and inaccuracy - is closed under the calculus. Two theorems showed completeness properties of the representations. We tied these concepts to a methodology in which measurement and recording of error knowledge may be institutionalized for persistent data - the calculus then using existing error measures to compute the impact of error on each data set retrieved. Such a methodology could apply in many database settings.

Example 4.1 (Scio-econ Data Co.): We work with one of the world's largest socio-economic data providers. They process thousands of data types from hundreds of data sources. Of several thousand customer help line calls per year, many involve data error. *What can be done if an error is suspected?* Sometimes a customer suggests a replacement datum based on a second data source. Where data are confirmed as incorrect, new data may be substituted. The problem here is that such a substitution modifies the source data, and it requires a judgment that a substitute datum is better than the original. Such judgments are often beyond the purview of a data administrator. In this organization, textual entries in a log are used to document experience with suspicious or incorrect data. An example text is: "*prices are wrong for several Belgian companies*". These text entries, however, are informal (e.g., not mathematically processable) and are not quantitative. ◆

Historically, personal familiarity with data allowed users to account - as best they could - for known patterns of error. Today, we believe that knowledge about data error should be a shared resource, e.g., a value-added addendum to data. Representations of error that are linked to the data dictionary and propagate through

operations on data would be useful in this organization so that the importance of error could be better known. This approach could allow an understanding - e.g., a quantitative mathematical model - of error to be accrued over time, by individual analysts, without modifying source data, and possibly characterizing error from each individual user's perspective - based on application context and error sensitivity.

We view this work only as a first step. For example, it may be possible to change or refine the mathematical construction of error. We believe, however, that a point space formalism and uncertainty framework, with properties similar to ours, should be provided for any model of data error. Economical ways of measuring and computer-encoding such error information, as well as complexity analysis of the calculus, are also necessary. Though not yet implemented in software, our model also makes no assumptions or requirements about change to underlying database systems, so its use in practice seems workable. We are currently implementing the (logical) error model in software so that some of these ideas may be tested. Error triples will be populated by inter-database comparisons and by maintenance of error tuples.

Example 4.2 (University Data Warehouse): We have worked with the architect of a Data Warehouse for a major university. He indicates that the representation and storage of organizational knowledge about data error is important to his project. Data in his warehouse come from "anonymous" sources, so potential users know little about data reliability. He fears that users will run analyses without understanding data error. Then, as errors are discovered, the warehouse project overall may lose credibility. ◆

The value of data sharing networks will be greatly reduced without rating and certification of data and without a means for communicating about error from those who know (e.g., administrators and prior users) to those who do not. Error-based filtering or retrieval of data may exist as a network mediation function when user error sensitivities are made explicit. Value-added network services for correcting or adjusting data (e.g., probabilistically) also seem desirable. Though documentation of data for semantic interoperability has focused on the meaning of data, a database with bad data may be worse than no database at all!

At VLDB 1993, one invited talk and two panels raised data quality as an important research area [18, 21, 23]. Clearly error is fundamental to data quality. Although database answers are generally wrong, there has been little done from a research standpoint to answer "by how much". This research is useful as a theory in that it leads to methods and tools for data quality administration in which (1) administrators have a framework and quantitative representation in which to record

information about error, (2) users get a better sense of data reliability and can evaluate the appropriateness of data for their application, and (3) the benefit to users of data quality improvement efforts can be known directly via the calculus, so that resources for data quality enhancement may be effectively applied.

5. REFERENCES

[1] Abiteboul, S., Hull, R., & Vianu, V. Foundations of Databases. Addison-Wesley, Reading, 1995.
[2] Atkinson, K. E. Elementary Numerical Analysis. John Wiley & Sons, New York, 1985.
[3] Barabba, V. P. Proceedings of the 1980 Conference on Census Undercount., Arlington, Va., 1980.
[4] Barbara, D., Garcia-Molina, H., & Porter, D. The Management of Probabilistic Data. *IEEE Transactions on Knowledge and Data Engineering, 4,* 5: 487-502, 1992.
[5] Ben-ari, M. Mathematical Logic for Computer Science. Prentice Hall, Int'l, Englewood Cliffs, 1993.
[6] Biemer, P., & Stokes, S. L. Approaches to the Modeling of Measurement Error. In*Measurement Errors in Surveys,* John Wiley & Sons, New York, 1991.
[7] Bunge, M. Semantics I: Sense and Reference. D. Reidel Publishing Company, Boston, 1974.
[8] Cavallo, R., & Pittarelli, M. The Theory of Probabilistic Databases. *Proceedings of the Thirteenth International Conference on Very Large Databases:* 71-81, Brighton, England, 1987.
[9] Farley, J. U., & Howard, J. A. Control of "error" in market research data. Lexington Books, Lexington, MA, 1975.
[10] Fuller, W. A. Measurement Error Models. John Wiley & Sons, New York, 1987.
[11] Grimmett, G. R., & Stirzaker, D. R. Probability and Random Processes. Clarendon, Oxford, 1982.
[12] Imielinski, T., & Lipski, W. Incomplete Information in Relational Databases. *JACM, 31,* 4, 1984.
[13] Johnson, J. R., Leitch, R. A., & Neter, J. Characteristics of Errors in Accounts Receivable and Inventory Audits. *Accounting Review, 56,* April: 270-293, 1981.
[14] Kennedy, P. A Guide to Econometrics. MIT Press, Cambridge, Mass, 1985.
[15] Kon, H. B., Madnick, S. E., & Siegel, M. D. (1995). Data Error: a Model for Representation and Propagation in Databases. MIT Composite Information Systems Laboratory. Internal WP.
[16] Laudon, K. C. Data Quality and Due Process in Large Record Systems. *CACM, 29,* 1, 1986.
[17] LePage, N. Data Quality Control at United States Fidelity and Guaranty Company. In G. E. Liepens & V. R. R. Uppuluri (Ed.), *Data Quality Control: Theory and Pragmatics*: 25-41, Marcel Dekker, Inc., New York, 1990.
[18] Madnick, S. E. The Voice of the Coustomer: Innovative and Useful Research Directions. *19th International Conference on Very Large Databases:* 701-704, Dublin, Ireland, 1993.
[19] Morey, R. C. Estimating and Improving Quality of Information in the MIS. *CACM, 25,* May, 1982.
[20] Motro, A. Integrity = Validity + Completeness. *ACM TODS, 14,* 4, 1989.
[21] Patterson, B. Panel Discussion: The Need for Data Quality. *19th International Conference on Very Large Databases:* 709, Dublin, Ireland, 1993.
[22] Reddy, M. P., & Wang, R. Y. A Data Quality Algebra for Estimating Query Result Quality. *Forthcoming, CISMOD Conference,* Bombay, 1996.
[23] Selinger, P. G. Predictions and Challenges for Database Systems in the Year 2000. *19th International Conference on Very Large Databases:* 667-675, Dublin, Ireland, 1993.
[24] Ullman, J. D. Principles of Database Systems. Computer Science Press, Rockville, MD, 1982.
[25] Wong, E. A Statistical Approach to Incomplete Information in Database Systems. *ACM TODS, 7,* 3: 470, 1982.[