# A Survey on Uncertainty Management in Data Integration

MATTEO MAGNANI and DANILO MONTESI
University of Bologna

In the last few years, uncertainty management has come to be recognized as a fundamental aspect of data integration. It is now accepted that it may not be possible to remove uncertainty generated during data integration processes and that uncertainty in itself may represent a source of relevant information. Several issues, such as the aggregation of uncertain mappings and the querying of uncertain mediated schemata, have been addressed by applying well-known uncertainty management theories. However, several problems lie unresolved. This article sketches an initial picture of this highly active research area; it details existing works in the light of a homogeneous framework, and identifies and discusses the leading issues awaiting solutions.

Categories and Subject Descriptors: H.2.5 [**Database Management**]: Heterogeneous Databases

General Terms: Theory

Additional Key Words and Phrases: Data integration, uncertainty

**ACM Reference Format:**
Magnani, M. and Montesi, D. 2010. A survey on uncertainty management in data integration. ACM J. Data Inform. Quality 2, 1, Article 5 (July 2010), 33 pages.
DOI = 10.1145/1805286.1805291. http://doi.acm.org/10.1145/1805286.1805291.



## 1. INTRODUCTION

Data integration is the process of providing the user with a unified view of data residing at different sources. Examples are relational databases, ontologies, and XML repositories [Lenzerini 2002]. In this work we focus on automated data integration, a difficult activity prone to errors [Gal 2006b]. One of the basic data integration tasks consists of comparing local data sources to identify *matching entities*, for example, two columns with `addr` and `home-ad` from

Authors' addresses: M. Magnani, Department of Computer Science, University of Bologna; email: matteo.magnani@cs.unibo.it; D. Montesi, Department of Computer Science, University of Bologna; email: montesi@cs.unibo.it.

two company databases both containing customer addresses. The information about matching entities, usually called *mapping*, is then used to merge the input data sources by including, for example, all of the customers' addresses into a single column. Several research efforts have led to the development of effective approaches to improve the accuracy of matching—for an overview of these methods, see Rahm and Bernstein [2001], Halevy et al. [2006b], and Euzenat and Shvaiko [2007]. However, automated tools may still fail in identifying all the correct mappings, for example, because of variations in columns. This may lead to errors in the merged database such as related data spread over several columns or single columns mixing heterogeneous information.

One way to tackle this problem is to explicitly represent the uncertainty generated by the data integration system and to consider it an important result of the integration process. Uncertainty is a state of limited knowledge, where we do not know which of two or more alternative statements is true. In the context of data integration, a typical example of uncertainty concerns the matching of objects, such as the fact that columns `addr` and `home-ad` may match or not. Keeping these alternatives leads to the production of multiple integrated databases, one for each choice. While traditional data integration methods more or less explicitly consider uncertainty as a problem, as something to be avoided, recent approaches treat uncertainty as an additional source of information, sometimes that is precious and that should be preserved.

In addition to uncertainty about the relationships of independent data sources, uncertainty may also affect other steps of a data integration process, as suggested in Sarma et al. [2008b]. The latter summarizes the results presented in [Dong et al. 2007; Sarma et al. 2008a]. *Data* may be uncertain, for example, in the event of data collected using information extraction systems or from unreliable data sources. The *mediated schema* may be uncertain due to the automated comparison of the data sources; for the same reason the *mappings* between the mediated schema and the data sources may also be uncertain. Finally, *queries* can be uncertain, because of the difficulty in expressing structured requirements when there is a lack of certain mediated schema. As well as these levels of uncertainty that may be present in a working data integration system, intermediate uncertain results can be produced during the *matching* step of the process, for example, the output of a single matcher comparing a pair of entities or an aggregated relationship produced by a pool of matchers. Therefore, when focusing on *uncertainty management in data integration* we refer to the comparison of uncertain data sources, to the generation of uncertain mappings between different data sources or between data sources and a mediated schema, to the representation of uncertainty in the mediated schema and to the execution of uncertain queries.

### 1.1 Motivation

The high interest in uncertainty in data integration is motivated by several data integration applications in which uncertainty is unavoidable. A prime class of applications are public online repositories, for example, Google Base, where anyone may upload structured content and where we cannot expect to

identify all the correct mappings. Similar problems can be found in data integration systems for the deep Web. These are characterized by a large number of sources, local data that is not easily available and schemata to be extracted from HTML pages. A second class of applications regards scientific data repositories such as bioinformatics databases where we do not necessarily know the exact mappings [Louie et al. 2007; Sarma et al. 2008b]. A third class of applications concerns the integration of data sources that are in themselves uncertain. Two relevant examples are databases with sensor data and databases built with information extracted automatically from the Web such as intelligence data. Future data integration systems will support a *pay-as-you-go* integration policy. Here newly added data sources will be available through simple keyword queries expressing uncertain requirements [Halevy et al. 2006a]. Finally, uncertainty has now been recognized as a typical result of the integration of geographical information systems given that data arising from observations of spatial entities are often imperfect [Worboys and Clementini 2001; Olteanu et al. 2008].

Data integration and uncertainty management are central topics in the field of data quality. Data integration is one of the basic activities used to improve the quality of data distributed among independent data sources. This is because it can both reduce its structural and semantic heterogeneity and redundancy, and increase its availability and degree of completeness (see e.g., the chapter on data integration by Batini and Scannapieco [2006]). At the same time, uncertainty is a kind of imperfection: one of the objectives of data quality processing is to reduce the amount and impact of imperfect data.

### 1.2 Methodology

This article includes a survey of the literature on uncertainty management in data integration. We queried the search engine Google Scholar, the IEEE Xplore database and the ACM digital library with all possible combinations of the keywords *data integration*/*schema integration* and *uncertain*/*ty, probability*/*stic* and *imperfect*/*ion* and considered at least the first 100 results for each query. These results were then filtered, we did this by examining the titles and abstracts of the papers so as to retain only those works that present data integration methods with an explicit representation of uncertainty aimed at producing uncertain results. We then checked the references of each selected paper, but no additional relevant publications were identified. Finally we looked at the Web sites of the most active researchers in the field to locate technical reports mentioned in collected papers in order to identify recent and as yet unpublished contributions. It is worth noticing that the papers reviewed in this survey appeared in the follow-up publications relating to some of the most important conferences and journals in the field of computer science such as SIGMOD, VLDB, ICDE, and the VLDB journal.

Not all the papers we looked at were included in the survey. We did not consider works where probabilities (or similarity degrees) were explicitly used only with the aim of making a choice between different alternatives, and so where uncertainty is lost during the data integration process. For example,

the system described in Hayne and Ram [1990] tries to assign probabilities to alternative relationships between pairs of schema objects. However, the characterizing feature of this and other early works using uncertainty theories is that after probabilities are evaluated a threshold is used to select matching and nonmatching objects. Therefore, the uncertainty generated during the integration process is lost. Similarly, probability theory has been used in instance integration (entity reconciliation or record linkage), and here again probabilities are used together with a decision model to choose exact mappings [Dey et al. 2002]. Another different application of probability theory in the field of data integration is described in Florescu et al. [1997]. In this case probabilities do not characterize the uncertainty in the matching process and in the integrated schema but are used to rank local data sources the aim being to improve query processing. Data is not uncertain and mappings between schema objects are well known.

At the end, we have identified about 30 papers dealing explicitly with data integration and uncertainty in addition to many other references mentioning the problem as relevant. Among the numerous papers considered we noticed that the majority of the approaches generating uncertain mediated schemata were published in 2005 and later, with the exception of a few visionary papers (Tseng et al. [1993] and Altareva and Conrad [2001; 2003], whose framework for uncertain data integration was later extended in Altareva and Conrad [2005]). In fact while uncertainty has always been included in data integration methods, only in the last few years that has been considered a valuable result in the process. This emerges quite clearly from the reviewed literature. In a 2003 survey, the problem of uncertain data management was not mentioned; it was stated that the main difficulty was the discovery of correct semantic relationships between schema objects [Halevy 2003]. Later, the problem of dealing with *imprecise mappings* was mentioned in another survey paper without explicit references to uncertainty management [Doan and Halevy 2005]. However, it was recognized that we will never be able to find all correct matches and that we should therefore be aware of possible errors and find ways to use partially incorrect results. That same year saw the publication of the first papers on uncertainty management in data integration; in a later survey, uncertainty management was explicitly referred to as one of the future challenges in the field [Halevy et al. 2006a].

### 1.3 Outline of the Article

The aim of this article is not just to provide a list of contributions. Here we attempt to show the reviewed papers as parts of a general and homogeneous data integration process, highlighting their mutual relationships and identifying what is missing. The aim is to encourage future research in this area.

Our article is organized as follows: in Section 2 we describe a generic process of data integration. That is a useful way of fitting the contributions in the survey into a homogeneous framework. Then in Section 3, we survey the main works dealing with uncertainty in data integration. We describe the main results, and classify existing approaches according to the step(s) they cover in the

Table I. The Main Recognized Classes of Data Imperfection

| Class | Example: John's tallness |
|---|---|
| No imperfection | 183 cm. |
| Absence/missing values | Not known. |
| Non–Specificity | Between 180 and 190 cm. |
| | 183 or 184 or 185 cm. |
| Vagueness | Not very tall. |
| Uncertainty | Perhaps, 183 cm. |
| Inconsistency | 183 and 184 and 185 cm. |
| Error | 170 cm. |

With examples about John's tallness

data integration process, the input and output data models, the features of data analyzed in matching the input data sources and the adopted theory of uncertainty. Finally, starting out from this systematization of existing proposals, we identify and discuss the key problems that remain open.

## 2. INTEGRATING DATA WITH UNCERTAINTY

In this section we first provide additional details about the concept of uncertainty, and in particular on the relationship between this concept and other kinds of data imperfection. Then, we split a general data integration activity into its main phases. This will serve as a sort of mediated schema for our survey: we will then provide mappings from the reviewed works to this general framework.

### 2.1 Uncertainty and Imperfection

So far, we have used the term *uncertainty*, as it is the most recurrent one in the works reviewed in the survey. However, it is worth noticing that in the database field, as well as in early works on uncertain data integration, it is well recognized that uncertainty is only one of many possible kinds of data imperfection that can affect a data source, as indicated in Table I [Bonissone and Tong 1985; Smithson 1989; Motro 1995; Demolombe 1997; Smets 1997; Pal 1999], adapted from Magnani and Montesi [2008a]. In this example perfect information is by convention set to 183 cm., and differing types of imperfection may coexist such as in: John is *probably not very tall*. The names we have assigned to different classes are used in many existing taxonomies of imperfection. However, there is no established consensus and other works have used slightly different classifications.

### 2.2 A Reference Data-Integration Process

In Figure 1 we have represented the main tasks constituting a general data-integration process. First, the input (also called *local*) data sources are translated into homogeneous data models (wrapping) to allow the comparison of otherwise heterogeneous information representation constructs. These homogeneous data representations may already contain uncertain information present in the original data sources or information generated during the wrapping phase if for example information extraction techniques are used. Then, these wrapped data sources are compared to each other using specialized

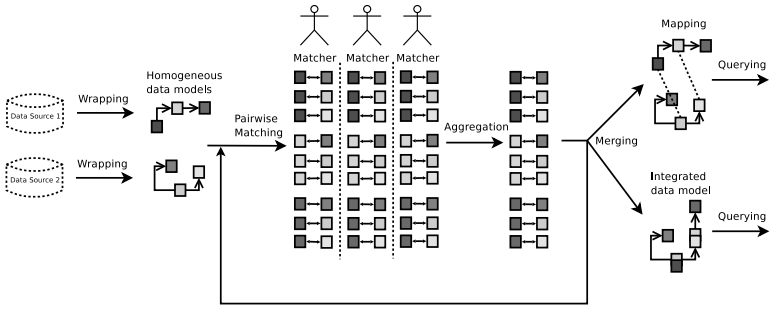


Fig. 1. A schematic view of a data integration process.

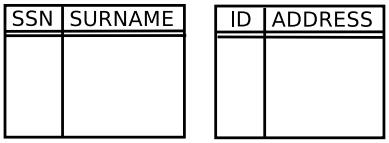


Fig. 2. Two relational tables representing independent data sources.

agents called *matchers*. A matcher can be a software agent as well as a human being, and usually each matcher implements one of the many matching algorithms available in the literature; for a survey of these approaches see Rahm and Bernstein [2001] and Euzenat and Shvaiko [2007]. The outcome of each matcher is combined with the others to produce an aggregated relationship between each pair of schema objects (aggregation). This process may involve several rounds of matching and aggregation and uses the result of each round to improve the next one. At this point the information is used to merge the local data sources into a global database to which we can ask queries. Global data sources may be materialized or implicitly represented using the mappings between the local data sources.

As an example of this process, consider the two data sources illustrated in Figure 2. The *wrapping* phase would generate some system-dependent internal representation of these tables—usually a graph with one node for each column and table, arcs representing their relationships and methods to fetch rows and metadata. In an uncertain data integration process, we allow the data contained in these tables to be uncertain, for example, one address could be `(6, Brown Street)` || `(9, Brown Street)` (the *or* operator (||) is used in Widom [2005] to represent alternative tuples in uncertain data models). Then several algorithms are used to *match* these structures. For example, the `SSN` and `ID` columns can be compared according to their names, their data types and their instances, that is, the actual values contained inside the columns. Uncertainty may also characterize the result of these algorithms leading for example to a probability of .8 for `SSN` and `ID` matching. Next the outcomes of the algorithms are *aggregated* again producing probabilistic relationships between the columns of the two tables. The joint information on all the relationships existing among the input data sources (*mapping*) is then used to generate a

single (virtual or materialized) database. This database may be also uncertain; for example, it may expose two alternative schemata (one for the case in which SSN and ID match, one for when they do not) and contain uncertain data as well (like the aforementioned address).

## 3. MANAGEMENT OF UNCERTAINTY

As we have indicated in the previous section, the goal of uncertain data integration is to use the uncertainty present in the data sources and/or generated during the matching phase, to build an uncertain integrated view of the data. This can then be seen as a compact representation of several alternative results of the process.

Uncertainty can be represented using quantitative methods, for example, specifying the probability that a mapping is correct, or qualitative methods, for example, using fuzzy sets and possibility theory to represent preferences about the correctness of a mapping. As we show in this section, quantitative models are the most frequently adopted in recent data integration methods, but also qualitative approaches are represented in the literature—usually with the aim of reducing the complexity of the manipulation of uncertainty.

In the following we review the main papers on uncertainty management in data integration with regard to the general process described in Section 2. In Table II we indicate the steps of the process it addresses for each main approach. "*I*" means that the step is not explicitly analyzed in the paper(s), but is supported by the adopted theoretical framework. For instance, Nottelmann and Straccia [2007] produce probabilistic–Datalog rules for which query processing was studied in Fuhr [1995]. However, they do not address it in the specific context of data integration. "*P*" indicates that the topic has been partially addressed, either by example or with oversimplifying assumptions. In the same summary table, we also indicate the theory of uncertainty used in the paper and the presence of experimental results. Some works cited in the body of the article have not been included in this table—in particular papers not providing new approaches for these specific steps and papers presenting case studies [Louie et al. 2007; Mimno et al. 2007].

### 3.1 Wrapping Uncertain Data

Data sources may be uncertain for many reasons: the source may have been extracted automatically by an unstructured data repository (such as the Web), it may have been collected using sensors or produced by a scientific experiment; its uncertainty could simply be due to the fact that it comes from untrustworthy sources. Although not directly related to data integration, there is much literature on data models for imperfect data: missing values [Codd 1979; Witold Lipski 1979], uncertainty [Barbara et al. 1992; Lee 1992; Tseng et al. 1993; Pittarelli 1994; Dey and Sarkar 1996; Lakshmanan et al. 1997; Fuhr and Rölleke 1997; Re et al. 2007; Sarma et al. 2006; Boulos et al. 2005; Cheng et al. 2005; Widom 2005; Agrawal et al. 2006] and vagueness [Bosc and Prade 1996] in relational databases. Van Keulen et al. [2005] suggest that semi-structured data models are well suited to representing the uncertainty produced during

Table II. Steps in a Generic Data Integration Process Addressed by the Cited Works

| | Wr | Ma | Ag | Me | Qu | Theory | Tests |
|---|---|---|---|---|---|---|---|
| [Tseng et al. 1993] | P | | | P | Y | Probability | |
| [van Keulen et al. 2005] | P | | | Y | P | Probability | Y |
| [Nottelmann and Straccia 2005; 2007] | P | Y | Y | I | I | Probability | Y |
| [Magnani et al. 2005] | | Y | Y | Y | | Evidence | |
| [Calì and Lukasiewicz 2006; 2008; Calì et al. 2008] | P/I | | | Y | P/I | Probability | |
| [Hunter and Liu 2006b, 2006a] | P | | | Y | | Probability, Evidence, Possibility | |
| [Gal et al. 2005; Gal 2006b; 2008; Roitman et al. 2008] | | Y | | | | Fuzzy | Y |
| [Wang et al. 2007] | | Y | Y | | | Evidence, Possibility | Y |
| [Nagy et al. 2007] | | Y | Y | | | Evidence | Y |
| [Marie and Gal 2007] | | Y | Y | | | Probability | Y |
| [Dong et al. 2007] | | | | Y | Y | Probability | |
| [Pankowski 2008] | | | | Y | | Probability | |
| [de Keijzer et al. 2006; de Keijzer and van Keulen 2007; 2008] | | | | Y | P | Probability | Y |
| [Magnani and Montesi 2008b] | | Y | Y | Y | P | Possibility/ Bipolar preferences | |
| [Agrawal et al. 2008] | Y | | | | Y | Possible Worlds | |
| [Sarma et al. 2008a] | | | | Y | I | Probability | Y |
| [Magnani and Montesi 2009a] | | Y | Y | Y | P | Probability | Y |
| [Gal et al. 2009] | | | | Y | Y | Probability | Y |

Wr: Wrapping, Ma: Matching, Ag: Aggregation, Me: Merging, Qu: Querying,
Y: addressed topic, I: implicit treatment, P: partial treatment

data integration, because of their flexibility. Additionally, less structured data is more likely to be uncertain. Indeed, one of the first applications of uncertain data integration to a real problem was outlined in Louie et al. [2007] in the context of biological databases. The main proposals of semistructured probabilistic data models are described in [Dekhtyar et al. 2001; Nierman and Jagadish 2002; Al-Khalifa et al. 2003; Hung et al. 2003b, 2003a; Magnani and Montesi 2008a]. As these works do not deal with the data integration process, we will not provide additional details here. However, it is worth noticing that some of these models have been presented as tools to query data originating from different sources [Tseng et al. 1993; Boulos et al. 2005].

The integration of data sources containing uncertain data is mentioned in Magnani and Montesi [2007] and Dong et al. [2007] as an important problem, but as yet it has not been studied thoroughly. Some works define data models that support the representation of uncertainty to some extent: van Keulen et al. [2005] define probabilistic XML trees, and Magnani and Montesi [2008b] allow the ranking of instances extracted by input schema objects; however the most promising approaches from this viewpoint are certainly the ones described in Nottelmann and Straccia [2005; 2007] and Calì and Lukasiewicz [2006]. Here the authors use respectively probabilistic Datalog programs and probabilistic

description logic programs. Nonetheless, so far the only work that deals explicitly with this aspect is Agrawal et al. [2008], in which some formal properties of the relationships between two uncertain databases (represented using a *possible worlds* model) are defined. In particular, this report defines notions of containment and consistency in (uncertain) data sources.

While advanced academic prototypes for managing uncertain data have been available for some years, there are still some open problems regarding the integration of uncertain data. These mainly concern their applicability within a data integration setting.

(1) Local data sources are usually defined as views over the global schema (or vice versa), and it is therefore necessary to know what it means when an answer to a query over probabilistic data (i.e., a view) is contained in the global database or when two uncertain data sources contain inconsistent data. Agrawal et al. [2008] provide a first step in this direction, but we should redefine the entire framework as illustrated in Lenzerini [2002] and extend it to probabilistic data.
(2) At some point the integration of uncertain data will necessitate the comparison of uncertain tuples or attributes. From our survey, it appears that this point has not been covered so far.
(3) Previous research on uncertain data management has focused on uncertain data rather than on uncertain schemata. However, to integrate data sources that have been the result of a previous uncertain data integration process, we should also consider how to compare uncertain schema objects whose metadata is uncertain.

### 3.2 Matching Data Sources with Uncertainty

The *matching* phase of a data integration process is where uncertainty is usually generated. In fact, using automated matchers we can only compare syntactic features of the data such as the names of two columns. That may not provide enough information to determine if they match or not.

Let us consider two schemata $\overline{S}_1$ and $\overline{S}_2$. The basic task in a data integration process consists in comparing $S_1 \in \overline{S}_1$ and $S_2 \in \overline{S}_2$ to identify the *relationship* occurring between them.[1] Figure 3 shows some possible choices of relationships, but the majority of systems uses only *match* and *not match*.

*Definition* 3.1 *Semantic relationship*. Let $R$ be a set of mutually exclusive relationships, and $\overline{S}_1$ and $\overline{S}_2$ two schemata. A *semantic relationship* is a function $F : \overline{S}_1 \times \overline{S}_2 \times R \rightarrow \{0, 1\}$ such that:

$$F(S_1, S_2, r) = \begin{cases} 1 & \text{if } (S_1, S_2) \in r \\ 0 & \text{if } (S_1, S_2) \notin r \end{cases} \quad (1)$$

with $\sum_{r \in R} F(S_1, S_2, r) = 1$ (i.e., exactly one relationship is correct).

[1]We focus on one-to-one relationships, which are used in almost all the works reviewed in this article.

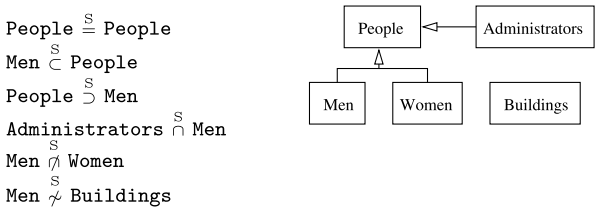


Fig. 3. A visual description of some possible semantic relationships between two entities. In this figure we have considered equivalence ($\overset{S}{=}$), subset-subsumption ($\overset{S}{\subset}$), superset-subsumption ($\overset{S}{\supset}$), overlapping ($\overset{S}{\cap}$), disjointness ($\overset{S}{\not\cap}$), and incompatibility ($\overset{S}{\not\sim}$). Other choices are *match* and *not match* ($M$, $\not M$), or ($=, \neq$), the latter mainly used in information fusion where single instances are compared instead of schema objects.

In the literature, this definition has been extended using probability theory, interval probabilities, conditional probabilities, fuzzy sets, and possibilistic logic, as we summarize in the following. The approach used by the papers reviewed in this survey is indicated in Table II.

Probabilistic methods model our ignorance about which relationship is correct, as defined in Dong et al. [2007]:

*Definition* 3.2 *Probabilistic semantic relationship*. Let $R$ be a set of mutually exclusive relationships, and $\overline{S}_1$ and $\overline{S}_2$ two schemata. A *semantic relationship* is a function $P : \overline{S}_1 \times \overline{S}_2 \times R \rightarrow [0, 1]$ such that:

—$P(S_1, S_2, r) \geq 0$
—$\forall(S_1, S_2)\ \sum_{r \in R} P(S_1, S_2, r) = 1$

*Example* 3.1 *Probabilistic semantic relationship*. Let $R = \{M, \not M\}$, $P(\text{Man}, \textit{Woman}, M) = .6$ and $P(\text{Man}, \text{Woman}, \not M) = .4$. P is a probabilistic semantic relationship, where Men and Women are more likely to match, with probability .6.

Magnani et al. [2005] suggest that indicating an exact probability for each alternative relationship may be difficult. Therefore, they introduce the notion of interval probabilistic relationship.

*Definition* 3.3 *Interval probabilistic semantic relationship*. Let $R$ be a set of mutually exclusive relationships, and $\overline{S}_1$ and $\overline{S}_2$ two schemata. A *semantic relationship* is a function $F : \overline{S}_1 \times \overline{S}_2 \times 2^R \rightarrow [0, 1]$ such that ($2^R$ indicates the powerset of $R$):

—$P(S_1, S_2, \emptyset) = 0$
—$\forall(S_1, S_2)\ \sum_{\overline{R} \subseteq R} P(S_1, S_2, \overline{R}) = 1.$

From this definition, we can compute the lower and upper probability of any relationship $r \in R$, as follows:

$$\text{LowerProbability}(S_1, S_2, r) = P(S_1, S_2, \{r\}) \tag{2}$$

$$\text{UpperProbability}(S_1, S_2, r) = \sum_{\overline{R} \subseteq R, r \in \overline{R}} P(S_1, S_2, \overline{R})\ . \tag{3}$$

Intuitively, $P(\{r\})$ is a *probability mass* supporting the relationship $r$. $P(\{r_1, r_2\})$ is a probability mass that supports either $r_1$ or $r_2$, but we do not know which one—therefore, $P(\{r_1, r_2\})$ will increase the upper bound of the interval probabilities of both $r_1$ and $r_2$.

*Example* 3.2 *Interval probabilistic semantic relationship*. Let $R = \{M, \not M\}$, $P(\text{Man, Woman}, \{M\}) = .6$ and $P(\text{Man, Woman}, \{M, \not M\}) = .4$. This assignment corresponds to the interval probabilities: $M : [.6, 1]$, $\not M : [0, .4]$.

An alternative approach developed in the field of information retrieval and first presented in Nottelmann and Straccia [2005] uses a probabilistic extension of datalog to encode uncertain relationships between schema objects. In this case, only binary relationships $(M, \not M)$ are used.

*Definition* 3.4 *Rule-based probabilistic semantic relationship*. Let $\overline{S}_1$ and $\overline{S}_2$ be two schemata. A semantic relationship is a rule:

$$\alpha S_1(d, v) \leftarrow S_2(d, v),$$

where $\alpha$ is the probability that for any document $d$ the value $v$ of attribute $S_2$ is also a value of attribute $S_1$.

This definition can be easily applied to databases instead of document metadata: if we think at $S_1$ as an attribute (e.g., the column of a relational table) in the mediated schema, and $S_2$ as an attribute in a local schema, then we can state that the probability of the values in $S_2$ to be also in $S_1$ is $\alpha$.

*Example* 3.3 *Rule-based probabilistic semantic relationship*. A rule:

$$.8\ \text{pubyear}(d,y) \leftarrow \text{year}(d,y)$$

means that 80% of the years appearing in a document $d$ correspond to the publication year of the document. This rule encodes the conditional probability $\text{P}(\text{pubyear}(d, y) \mid \text{year}(d, y))$.

In the definitions we have already encountered it is worth noting that there are two different interpretations of probabilities. The most common is to assume that there is just one correct relationship and that probabilities express our belief in each alternative (but again only one is correct in the real world). On the contrary, in the second interpretation, the one used in the rule-based approach holds that one relationship can be partially correct, such as in the case of publication years that correspond solely to some of the years stored in the documents. In Dong et al. [2007] these interpretations were therefore called *by-table* and *by-tuple* respectively. We shall provide additional details on the different computational complexity associated with these two interpretations later.

Another way to express partial relationships is to use fuzzy sets namely the main tool for formalizing the *similarity values* used in many traditional data integration methods. This approach is used in Gal et al. [2005], where the authors define an evaluation framework for approximate mappings (also discussed in Gal [2008]).

*Definition* 3.5 *Fuzzy semantic relationship*. Let $\overline{S}_1$ and $\overline{S}_2$ be two schemata. A semantic relationship is a function $\mu : \overline{S}_1 \times \overline{S}_2 \rightarrow [0, 1]$ representing the similarity of schema objects $S_1$ and $S_2$.

Finally, another nonprobabilistic approach is described in Magnani and Montesi [2008b], based on possibilistic logic and on the preference representation framework presented in Benferhat et al. [2006]. To simplify the production of uncertain relationships and to reduce the complexity of their manipulation, this work allows the expression of preferences.

*Definition* 3.6 *Preference-based semantic relationship*. Let $\overline{S}_1$ and $\overline{S}_2$ be two sets of schema objects. Also, let $L^-$ be a totally ordered set of rejection levels, with $\min(L^-)$ indicating indifference and $\max(L^-)$ strong rejection, and let $L^+$ be a totally ordered set of satisfaction levels, with $\min(L^+)$ indicating indifference and $\max(L^+)$ complete satisfaction. A *semantic relationship* is defined by a pair of functions $F_P^- : \overline{S}_1 \times \overline{S}_2 \rightarrow L^-$, representing a negative preference, and $F_P^+ : \overline{S}_1 \times \overline{S}_2 \rightarrow L^+$, representing a positive preference.

*Example* 3.4 *Preference-based semantic relationship*. Let $L^+$ be: (indifference, weak preference, medium preference and strong preference) and $L^-$ be: (not rejected, weak rejection, medium rejection and strong rejection). As an example, using these ranges, we can define the following preferences:

—$F_P^+$(People, Woman) = strong preference.
—$F_P^+$(Man, Administrators) = weak preference.
—$F_P^-$(Man, Woman) = not rejected.

In this way, we are encoding the fact that we support the matching of both the pair (People, Woman) and the pair (Man, Administrators), the first more than the second, and we do not reject the idea that (Man, Woman) match, even if we do not express any preference about it.

The most interesting feature of this definition is that it uses a symbolic approach, for example, if $l_1^+, l_2^+ \in L^+$, $F_P^+(S_1, S_2) = l_1^+$, $F_P^+(S_1, S_3) = l_2^+$ and $l_1^+ > l_2^+$, we can say that $(S_1, S_2)$ is preferred to $(S_1, S_3)$, but not *how much*. This is a major difference with respect to probabilistic systems and corresponds to a reduced expressiveness and a consequent reduced complexity as indicated in Section 3.5.

While we have shown that many alternative definitions of uncertain semantic relationships have been proposed, the main open problems regarding the comparison of data sources concern the generation of these relationships, that is, the assignment of probabilities, intervals or preferences. Each matcher must produce a measure of uncertainty from the comparison of some features of the data and this is notoriously delicate matter. For example, what is the probability that columns `home` and `house` match? This is a difficult question also for a human being.

Table III. Matchers and Aggregation of Matcher Outcomes

| | Rels | Matchers | Aggregation |
|---|---|---|---|
| [Magnani et al. 2005] | $\overset{S}{=}$, $\overset{S}{\subset}$, $\overset{S}{\supset}$, $\overset{S}{\cap}$, $\overset{S}{\emptyset}$, $\overset{S}{\not\sim}$ | cardinalities, names (wordnet, sub-strings), data statistics, data types, attribute values | Dempster's rule |
| [Nottelmann and Straccia 2005; 2007] | $M$, $\not{M}$ | names (precise comparison and stems), data type, values (with and without term , weighting) k-nearest neighbor, naive Bayes, KL-distance | Weighted sum of classifier estimates (i.e., expectation) |
| [Wang et al. 2007] | $M$, $\not{M}$ | edit distance, linguistic-based, structure-based | Dempster's rule (for beliefs), min (for possibilities) |
| [Nagy et al. 2007] | $M$, $\not{M}$ | not specified (Jaccard, Jaro-Winkler string distances, wordnet. . . ) | Dempster's rule |
| [Marie and Gal 2007] | $M$, $\not{M}$ | term, composition, precedence | Naive Bayes Heuristic |
| [Magnani and Montesi 2008b ] | $M$, $\not{M}$ | names (wordnet, string distances), data statistics, data types, attribute values | Union |
| [Magnani and Montesi 2009a] | $M$, $\not{M}$ | names (wordnet, string distances), data statistics, data types, attribute values | Dempster's rule |

### 3.3 Aggregation of Multi-Matcher Outcomes

The aggregation of the uncertain outcome of different matchers is usually uncertain too, as a result of the uncertainty generated during the matching phase, and may involve some knowledge of the relationships between the matchers.

Today it is well recognized that no single matching algorithm outperforms all the others independently of the input schemata. This has led to the definition of data integration architectures made of many *matchers* as in Do and Rahm [2007], where each matcher implements a different algorithm. This architecture has many desirable features: its implementation can be focused on simple matchers, it is scalable as matchers can be easily removed and added, and it is easily parallelizable.

However, if we want a sound representation and manipulation of uncertainty, this architecture poses some additional challenges.

(1) The output of the matchers must be expressed using the adopted uncertainty management theory.
(2) We need a way to aggregate the outcomes of the matchers.
(3) We can no longer add and remove matchers without considering their mutual relationships.

In this section we discuss how these challenges have been addressed in the literature. Table III indicates the matchers used in the reviewed approaches,

the relationships they try to identify, and the aggregation method; we have indicated only those papers providing details about the adopted matchers.

With regard to the first challenge, probabilistic methods are under criticism because it is not easy to justify the produced values. In the papers under analysis, ad hoc methods not grounded in theory are often used, and different interpretations of the theory such as subjective or frequentistic ones, are adopted in different matchers. This problem also affects interval probabilistic approaches to a minor degree. Qualitative methods are certainly more intuitive, but it is still difficult to ensure that different matchers adopt the same ranges of values.

The aggregation of the probabilistic outcomes of different matchers is another complex problem. Magnani et al. [2005] use Dempster's aggregation rule in the context of interval probabilistic matching. This rule produces a valid interval probability distribution as long as the matchers do not contradict each other.

*Definition* 3.7 *Aggregation of interval prob. semantic relationships*. Let $R$ be a set of mutually exclusive relationships, and $\overline{S}_1$ and $\overline{S}_2$ two schemata. Also, let $P_{M_1}$ and $P_{M_2}$ be the interval probabilistic semantic relationships produced by two matchers $M_1$ and $M_2$. For all $S_1 \in \overline{S}_1$, $S_2 \in \overline{S}_2$, and $\overline{R} \subseteq R$, the aggregation is defined as:

$$P(S_1, S_2, \overline{R}) = \begin{cases} 0 & \text{if } \overline{R} = \emptyset \\ \frac{\sum_{\overline{R_1} \cap \overline{R_2} = \overline{R}} P_{M_1}(S_1, S_2, \overline{R_1}) P_{M_2}(S_1, S_2, \overline{R_2})}{1 - \sum_{\overline{R_1} \cap \overline{R_2} = \emptyset} P_{M_1}(S_1, S_2, \overline{R_1}) P_{M_2}(S_1, S_2, \overline{R_2})} & \text{if } \overline{R} \neq \emptyset \end{cases}. \quad (4)$$

Intuitively, this equation defines the following computation: if a set of relationships $\overline{R_1}$ is supported by $P_{M_1}$, and another set of relationships $\overline{R_2}$ is supported by $P_{M_2}$, then the intersection of $\overline{R_1}$ and $\overline{R_2}$ is supported by both. For example, if a matcher supports the relationships $\{\overset{S}{\supset}, \overset{S}{=}\}$ and another matcher supports $\{\overset{S}{\cap}, \overset{S}{=}\}$, the combination will support the common relationship $\overset{S}{=}$. The denominator performs a normalization to make the function P sum to 1.

The combination of $n$ matchers is obtained by iteratively applying this rule $n - 1$ times. The complexity of *exact methods* for performing Dempster's combination rule is exponential on the number of relationships, because it must consider all its subsets in the worst case. Therefore, as we deal with at most six alternative relationships, the complexity of the combination is bounded by a small constant. However, the main problem of this rule is that the result of the computation is correct only if the outcomes of the matchers are probabilistically independent. Unfortunately this is rarely the case: whenever two matchers analyze the same features of two schema objects, for example, their names, their outcomes are very likely to be correlated. Dempster's rule has also been used extensively in the field of Semantic Web and Ontology Matching [Wang et al. 2007; Nagy et al. 2007], where it suffers from the same problem.

It is worth noticing that this rule can also be used for simple probabilistic relationships in which case it reduces to the following:

*Definition* 3.8 *Aggregation of probabilistic semantic relationships*. Let $R$ be a set of mutually exclusive relationships, and $\overline{S}_1$ and $\overline{S}_2$ two schemata. Also, let $P_{M_1}$ and $P_{M_2}$ be the probabilistic semantic relationships produced by two

matchers $M_1$ and $M_2$. For all $S_1 \in \overline{S}_1$, $S_2 \in \overline{S}_2$, and $r \in R$, the aggregated probability is defined as:

$$P(S_1, S_2, r) = \frac{P_{M_1}(r)P_{M_2}(r)}{1 - \sum_{r' \neq r''} P_{M_1}(r')P_{M_2}(r'')} \tag{5}$$

An alternative approach based on matcher weighting is proposed in Nottelmann and Straccia [2005; 2007]:

*Definition* 3.9 *Aggregation of rule-based semantic relationships*. Let $\alpha$ T(d,y) $\leftarrow$ S(d,y) be a rule-based semantic relationship, let $P(M_k)$ indicate the probabilistic weight of matcher $M_k$, and $P(S|T, M_k)$ be the probability estimated by $M_k$.

$$P(S|T) \approx \sum_{k=1}^{n} P(S|T, M_k) \cdot P(M_k) \tag{6}$$

Then, Bayes' theorem is used to obtain $P(T|S) = P(S|T)\frac{P(T)}{P(S)}$. This definition is based on an underlying assumption that only one of the matchers is correct; its computational complexity is linear in the number of matchers.

So far, we have mentioned probabilistic aggregation methods that do not consider probabilistic dependencies. There is still no conclusive experimental evidence highlighting the impact of dependencies on the result of these aggregations. However, some preliminary tests suggest that they may negatively affect the correctness of the resulting relationships. Marie and Gal [2007] model matcher aggregation as a classifier learning activity. In this work, the output of each matcher is considered as a $|\overline{S}_1| \cdot |\overline{S}_2|$ matrix, where the entry $(i, j)$ contains the probability of attribute $i$ from schema $\overline{S}_1$ to match attribute $j$ from schema $\overline{S}_2$. These matrices are regarded as training sets, and a Naive Bayesian classifier is learned from them. The experiments reported in the paper, although preliminary, show that the independence assumption, which is not satisfied by the matchers used in the experiments, may have a negative impact on the precision and recall of the merged matcher.

When we use a qualitative approach, the aggregation of the outcome of different matchers can be simplified. In particular, Magnani and Montesi [2008b] define the combination of preferences expressed by $n$ matchers as their (bag) union and state that any other triangular conorm can be used [Klement et al. 2000].

*Example* 3.5. Let three matchers express the following positive preferences about a pair of schema objects:

$M_1$ Strong preference.
$M_2$ Strong preference.
$M_3$ Weak preference.

The aggregated expression of preferences will then be: {(Strong preference, 2), (Weak preference, 1)}. Using the more typical `max` function, the aggregation would result in a single Strong preference.

The union of positive and negative preferences may generate inconsistencies, that is, opposite opinions of different matchers. In this case, matchers that reject an option have greater priority. Using this technique, the time needed to aggregate the outcomes of the matching phase is linear in the number of matchers. However, there are still no experimental results that assess its effectiveness or which compare the impact of using alternative T-conorms on the precision and recall of the method.

The main open problems regarding the aggregation of the matcher outcomes concern their mutual relationships.

(1) The probability generated by different matchers should be compatible, that is, produced according to a common interpretation; otherwise the influence of some matchers which tend to output high values may limit the contribution of others. In the reviewed literature no formal concept of *compatible uncertainty measures* has been defined or discussed.
(2) Even when compatible uncertain relationships have been produced, their aggregation should consider probabilistic dependencies between the matchers.
(3) The problem of dynamic pools of matchers that change their organization according to the generated uncertain semantic relationships has been only partially addressed [Magnani and Montesi 2007; de Keijzer and van Keulen 2007]. For example in Magnani and Montesi [2007] the generation of inconsistent results is used to identify portions of the data sources where some matchers do not perform well.

### 3.4 Merging Data Sources Based on Uncertain Mappings

As for the aggregation of the outcomes of the matchers, the source of the uncertainty that may affect a mapping still comes from the matching phase although it is manipulated to merge the result of each matcher.

There are two main approaches to perform a merging. One option is to materialize the integrated database. This is typical with small data sources like single XML documents with a shared XML schema. This activity is often called *information fusion* and there is much literature available about it. The other option is to use one or more mediated schemata to translate global queries into queries on local data sources—for example, the integrated database is implicitly defined by the mediated schema and the mappings—this is typical for large distributed databases. In this case the mediated schema may be provided as input to the data integration process or it may be produced using the information collected during the matching phase. The two approaches are also called respectively *data* and *schema* integration in the literature when the term *data integration* is not ambiguous.

In Table IV we indicate the most significant merging approaches reviewed together with the adopted data model, the set of relationships used in the mapping, the method used to obtain the mapping from the result of the matching step (detailed below), and whether it is a case of schema or data integration.

Table IV. Data Models and Merging Approaches

| | Data Model | Rels | Merging | Kind |
|---|---|---|---|---|
| [Tseng et al. 1993] | Relational | Probabilistic partial values | Generalized union | Schema Data |
| [van Keulen et al. 2005] | p-XML | $=, \neq$ | All combinations, constrained by XML schemata, and probabilities assigned with a frequentistic interpretation | Data |
| [Nottelmann and Straccia 2005; 2007] | Sets of schema objects | $M, \not{M}$ | Union of all p-datalog rules | Schema |
| [Magnani et al. 2005] | ER | $\overset{S}{=}, \overset{S}{\subset}, \overset{S}{\supset}, \overset{S}{\cap}, \overset{S}{\not\cap}, \overset{S}{\not\sim}$ | All combinations, excluding mutually exclusive | Schema |
| [Calì and Lukasiewicz 2006; 2008; Calì et al. 2008] | Description Logics | $M, \not{M}$ | Not specified | Schema |
| [Hunter and Liu 2006b, 2006a] | XML (with probability, belief, and possibility functions) | $M, \not{M}$ | Dempster's rule, min, max | Data |
| [Dong et al. 2007] | Relational | $M, \not{M}$ | Certain mediated schema | Schema |
| [Pankowski 2008] | XML | $M, \not{M}$ inside path expressions | Not specified | Schema |
| [Magnani and Montesi 2008b] | Sets of schema objects | $M, \not{M}$ | Clusters of matching objects | Schema |
| [Agrawal et al. 2008] | Possible worlds | $M, \not{M}$ | Certain mediated schema | Schema |
| [Sarma et al. 2008a] | Single relational table | $M, \not{M}$ | Construction of a probabilistic mediated schema from similarity of source attributes and co-occurrence constraints | Schema |
| [Magnani and Montesi 2009a] | Sets of schema objects | $M, \not{M}$ | Probabilistic clusters of matching objects | Schema |
| [Gal et al. 2009] | Relational | $M, \not{M}$ | Certain mediated schema (queries on single tables) | Schema |

3.4.1 *Implicit Merging.* This kind of merging basically corresponds to the generation of an uncertain mapping and, if it is not already available, of the mediated schema. Therefore for each alternative approach we will provide the corresponding definition of mapping and describe how to obtain it. The following definitions look very much like the definitions of simple relationships. However, mappings may be much more complex: they involve many relationships at the same time and the presence of dependencies between them makes it hard to compute their probability.

A mapping associates each pair of objects from the two input data sources to a semantic relationship like the ones represented in Figure 3. Uncertain mappings are defined as follows in Magnani et al. [2005].

*Definition* 3.10 *Interval probabilistic schema mapping*. Let $R$ be a set of mutually exclusive semantic relationships, and $\overline{S}_1$ and $\overline{S}_2$ two sets of schema objects. A mapping is a function $m : \overline{S}_1 \times \overline{S}_2 \to R$. We indicate with $M(\overline{S}_1, \overline{S}_2, R)$ the set of all possible mappings. An *interval probabilistic schema mapping* is a tuple $(\overline{S}_1, \overline{S}_2, R, P)$ where $P : 2^{M(\overline{S}_1, \overline{S}_2, R)} \to [0, 1]$ is a function over the powerset of all mappings between $\overline{S}_1$ and $\overline{S}_2$ such that:

—$P(\emptyset) = 0$

—$\sum_{\overline{M} \subseteq M(\overline{S}_1, \overline{S}_2, R)} P(\overline{M}) = 1$.

As in the case of simple relationships, we can compute the lower and upper probability of any mapping as follows.

$$\text{LowerProbability}(m) = P(m) \tag{7}$$

$$\text{UpperProbability}(m) = \sum_{\overline{M} \subseteq M(\overline{S}_1, \overline{S}_2, R), m \in \overline{M}} P(\overline{M}) \ . \tag{8}$$

This definition is reduced to the definition of probabilistic mapping when the domain of the function $P$ is the set of mappings instead of its powerset:

*Definition* 3.11 *Probabilistic schema mapping (1)*. Let $R$ be a set of mutually exclusive semantic relationships, and $\overline{S}_1$ and $\overline{S}_2$ two sets of schema objects. A mapping is a function $m : \overline{S}_1 \times \overline{S}_2 \to R$. We indicate with $M(\overline{S}_1, \overline{S}_2, R)$ the set of all possible mappings. An interval probabilistic schema mapping is a tuple $(\overline{S}_1, \overline{S}_2, R, P)$ where $P : M(\overline{S}_1, \overline{S}_2, R) \to [0, 1]$ is a function over the set of all mappings between $\overline{S}_1$ and $\overline{S}_2$ such that:

—$P(\emptyset) = 0$

—$\sum_{m \in M(\overline{S}_1, \overline{S}_2, R)} P(m) = 1$.

It is easy to see that both definitions admit a number of possible schema mappings. This is exponential in the number of pairs of schema objects, and the probability of each mapping is a product of conditional probabilities. For example, the matchers may assign some probability to the events that schema objects A and B are equivalent, that schema objects B and C are equivalent as well, but A and C are incompatible ($A \overset{S}{\equiv} B$, $B \overset{S}{\equiv} C$, $C \overset{S}{\not\equiv} A$). Evidently the probability of this mapping would be 0 and not the product of the probabilities locally assigned to the three relationships. Magnani and Montesi [2007] use a top-k algorithm that only considers this kind of dependency and whose worst-case complexity is still exponential in the number of pairs of schema objects. In addition, the probabilities it returns are not correct in the presence of other probabilistic dependencies.

Dong et al. [2007] provide the first formal analysis of probabilistic mappings. In their work, a probabilistic schema mapping is a set of possible (ordinary) mappings between a source schema and a target schema, where each possible mapping has an associated probability.

*Definition* 3.12 *Probabilistic schema mapping (2).* Let $\overline{S}$ and $\overline{T}$ be two relational schemata. A probabilistic mapping pM is a triple (S,T,**m**), where S is a relation in $\overline{S}$, T is a relation in $\overline{T}$, and **m** is a set $\{(m_1, Pr(m_1)), \ldots, (m_l, Pr(m_l))\}$, such that

—for $i \in [1, l]$, $m_i$ is a one-to-one mapping between S and T, and for every $i, j \in [1, l], i \neq j \Rightarrow m_i \neq m_j$.

—$Pr(m_i) \in [0, 1]$ and $\sum_{i=1}^{l} Pr(m_i) = 1$.

A probabilistic schema mapping is a set of probabilistic mappings between relations in $\overline{S}$ and in $\overline{T}$, where every relation in either $\overline{S}$ or $\overline{T}$ appears in at most one probabilistic mapping.

In this work, the authors assume the existence of a certain mediated schema. This assumption limits the application domain of these mappings to situations where the mediated schema is not obtained automatically from the analysis of the local sources.

Dong et al. [2007] suggest two semantics for probabilistic schema mappings corresponding to different complexity classes of query answering. Assume we have one relational table $O_1$, which matches table $O_2$ with probability $\frac{1}{2}$. In *by-table* semantics, the correspondence is either correct or not, we do not know. These semantics are basically the same as in Magnani et al. [2005]. In *by-tuple* semantics, the correspondence is partially correct: it is true for half of the tuples. As we shall see later, by-tuple semantics is intrinsically more complex than by-table semantics and this affects the computational complexity of query answering. In fact, with the former we need to consider all possible ways of assigning alternative mappings to the tuples. For instance if a table contains three tuples and there are two possible mappings $m_1$ and $m_2$, we can say that tuple 1 must be interpreted according to $m_1$ and the others according to $m_2$ ($\langle m_1, m_2, m_2 \rangle$), or that tuples 1 and 3 must be interpreted according to $m_1$ ($\langle m_1, m_2, m_1 \rangle$), and so on for all other alternatives. It would also be interesting to compare this interpretation with one based on fuzzy sets, which are well suited to representing this kind of vague relationship. These two semantics have been adopted by subsequent works studying aggregate query operators [Gal et al. 2009] and the integration of XML data [Pankowski 2008].

In the same work, the authors deal also with the compact representation of mappings. The representations proposed in the paper are consistent with what we would expect from a probabilistic approach: if the relationships between pairs of schema objects are independent of each other, the space complexity of probabilistic mappings can be reduced to a linear function of the number of pairs. However, in the general case, we need to represent all mappings one by one, with their associated probability.

The logic programming approach presented in Nottelmann and Straccia [2005; 2007] defines an uncertain mapping as a set of p-datalog mapping rules. In particular:

*Definition* 3.13 *Rule-based schema mapping*. Let **S** be a source schema and **T** a target schema. A mapping is a tuple (**T**, **S**, Σ), where Σ is a set of mapping rules.

Also in this case, the authors assume the existence of a target schema. Learning a schema mapping which here corresponds to the merging phase, consists in the following four steps.

(1) For each possible relationship T(d,v) ← S(d,v), estimate the probability $P(S|T)$ (as we have previously mentioned, this is done by aggregating the result of the matchers through a weighted sum using an independence assumption).
(2) Compute all sets Σ of mapping rules, and estimate its quality using these probabilities.
(3) Select the best schema mapping according to this estimation.
(4) Compute $P(T|S)$ for each mapping rule in the best Σ.

We will not repeat the details of the process of probability estimation as they can be found in the original papers. We only point out that here again the manipulation of probabilities suffers from an underlying assumption of independence, that is not satisfied in the majority of data models such as ER and OWL. In addition, we notice that the number of all sets Σ of mapping rules is exponential in the number of pairs of schema objects, making this method useful only for very small schemata.

3.4.2 *Generation of an Uncertain Mediated Schema.* Magnani and Montesi [2008b] following Magnani et al. [2005], try to generate the mediated schema starting from the input data sources. If two schema objects $S_1$ and $S_2$ do not match, that is, they are not related to each other, they will be inserted as separate objects into the mediated schema. On the contrary, if two local schema objects match, a third object will also be added to the mediated schema, containing instances from both input schema objects and representing the fact that the local objects are incomplete views over it. This is exemplified in Figure 4, where the schema object depicted on the right is generated because the two local `title` schema objects match each other.

A similar approach is adopted in Magnani and Montesi [2009a], but using a quantitative approach, and in Sarma et al. [2008a], where the authors discuss how to automatically obtain an uncertain mediated schema from the local sources by clustering together attributes with a high probability of matching, and producing alternative clusterings for mappings with a lower probability.

3.4.3 *Materialized Merging.* The option of explicitly building an integrated data source has been mainly used in methods that focus on data and not on schemata. These approaches are also known as *information fusion methods*.

Tseng et al. [1993] define an *integration* operator to compute the union of two input relations with conflicting values where all the alternatives are included in the result. For example, if the first input relation states that the age of a person is 23 and the second states that it is 24, the output relation will

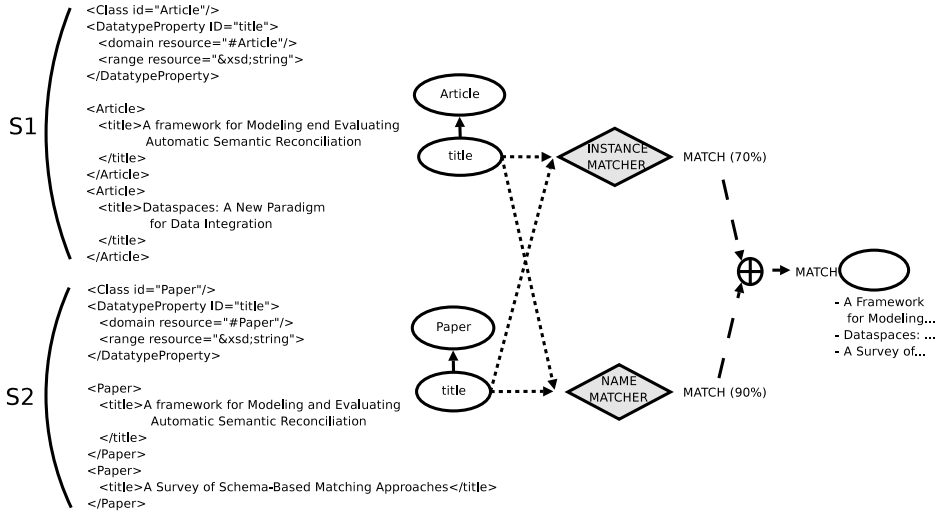


Fig. 4. An example of data integration with the generation of the mediated schema.

indicate the set {23,24}. In addition, these values would be annotated with probabilities, that without additional information will be equally distributed among the alternatives.

Van Keulen et al. [2005] assume the existence of a *matching engine* providing the relationships between different data entities. Here they define a data model that can be used to merge data trees. The authors also assume that the merged trees share the same XML schema. Therefore, the only source of uncertainty comes from elements with the same name and different text nodes. However, even with this simplifying assumption, the number of alternative possibilities obtained by merging sequences of text values is high. While improving the matching phase would significantly reduce this number, it appears that to use this approach it is necessary to reduce the cardinality of the solution space. An example of a merged tree is illustrated in Figure 5, taken from the original paper.

With regard to the way in which probabilities are assigned to different alternatives during merging, van Keulen et al. [2005] use a frequentistic approach. For instance, if we merge 100 data trees, and 99 of them have a `person` element with value John while one has a `person` element with value Jon, the probability assigned to John will be .99.

A similar approach is presented in Hunter and Liu [2006b; 2006a], where XML documents are enriched with annotations of probability, belief and possibility. The authors face the same complexity problems and tackle them using external domain knowledge to reduce the number of alternatives—as suggested too in van Keulen et al. [2005]. In addition they show how different measures of uncertainty such as beliefs and possibilities, may be merged together. This correspondence was already pointed out in Shafer [1976], but its meaningfulness is still to be verified. It is worth noting here that the application of Dempster's

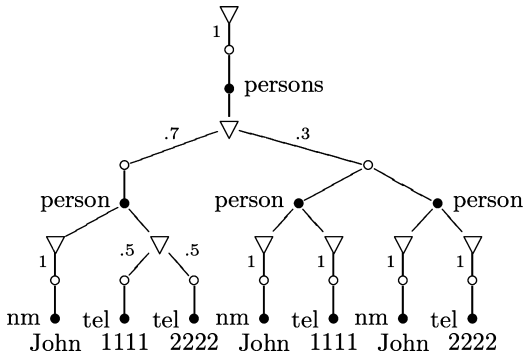


Fig. 5. An example of probabilistic XML tree, with XML (•), possibility (∇) and probability (○) nodes.

rule, used to perform the merging, suffers from the aforementioned problems in the event of probabilistic dependencies.

The main open problems regarding the generation of uncertain mappings once again concern the relationships between different schema objects that may cause probabilistic dependencies, and the difficulty in managing an uncertain schema.

(1) A mapping may contain relationships between common objects, for example, the relationship between columns `address1` and `address2` and the one between columns `address1` and `home-address`. These may influence our belief about the relationship between `address2` and `home-address` and thus the way in which we compute the joint probabilities.
(2) The subject of uncertain schemata as mentioned in this section has as yet only been treated by a few recent works so far and needs to be extended to other data models and tested in real applications. In fact it is still unclear how these uncertain schemata can be used in practice.

### 3.5 Querying Uncertain Mediated Schemata

Uncertainty may affect both the result of queries over uncertain mediated schemata and the query itself. In the first case, it is again a result of the matching phase. In the second case, it may come from the unavailability of mappings or of a well-defined mediated schema. Therefore users have to formulate queries without precise references to schema elements—such as keyword queries—and the system has to translate them into a set of alternative structured queries.

The first theoretical results concerning the complexity of query execution in uncertain data integration systems were published by Dong et al. [2007]. We remind the reader that these results focus on the by-table and by-tuple semantics introduced in the last subsection. The authors study the behavior of selection-projection-join queries and show that for by-table semantics (and also for two large classes of queries within the by-tuple semantics), query answering

is in PTIME both on the size of mappings and on the size of data. On the contrary, returning all the results of a query with by-tuple semantics is in general #P-complete. It is worth noticing that these complexity classes do not change even if we use more complex mappings, for example one-to-many relationships (like in: person → name + surname, or income → monthly_income*12). Obviously, while query answering keeps the same complexity, the matching phase not addressed in Dong et al. [2007] could become intractable because of the high number of possible combinations of sets of attributes. In addition, to apply these results to complete uncertain data integration applications, we need to consider that the size of mappings is in general exponential with regard to the number of schema objects in the local data sources (for instance, the number of columns and tables in a relational database). Therefore, we can ask these kinds of queries in an uncertain data integration system as long as the matching phase returns a small number of alternative mappings. These results have been extended for application also in the presence of an uncertain mediated schema in Sarma et al. [2008a] and to XML data in Pankowski [2008].

Agrawal et al. [2008] address the problem of answering queries posed over a mediated schema when the source databases are uncertain. In particular, as many mediated databases consistent with the sources are possible, there can also be many alternative query answers. The authors thus define two notions (correct answer and strongest correct answer) to characterize good and best query answers. Intuitively, an answer is good if it is contained in the answers of all mediated databases consistent with the sources.

Many works mention approaches to reduce the number of mappings, thus increasing the efficiency of the process. These approaches do not directly address the problem of query answering but may reduce the execution time of queries on the uncertain mediated schema. Nottelmann and Straccia [2007] suggest that some discovered rules can be removed by using ad hoc methods like thresholds, top-k, or limiting the number of rules for each target attribute. Gal [2006a] shows that the analysis of the top-k mappings can be used as a selection criterion—keeping the relationships that are more stable in high-likelihood mappings. Both Magnani et al. [2005] and van Keulen et al. [2005] suggest using thresholds and constraints to remove some possibilities; however checking these constraints may become an additional source of complexity. Also, Sarma et al. [2008a] propose removing some of the uncertainty: all mappings with a probability greater than a predefined threshold $(\tau + \epsilon)$ are considered as certain, and all mappings with a probability less than a related threshold $(\tau - \epsilon)$ are considered as wrong. De Keijzer and van Keulen [2007] suggest user feedback can be used to reduce the number of possible worlds. While uncertain data integration is of particular interest when there are no human users participating in the process, it is still possible to take advantage of human feedback at query time to update the source of information responsible for the wrong answer. Another kind of user involvement is suggested in Magnani and Montesi [2007] and supported by some preliminary experiments. When the system produces inconsistent results, such as unsatisfiable ER schemata, it can also identify the combinations of relationships that generated the inconsistency and ask the user to check them. Finally, de Keijzer et al.

[2006] and de Keijzer and van Keulen [2008] use consistency rules that remove part of the possible worlds generated during the matching step.

While these papers present preliminary experimental results which show an effective reduction of the number of possible worlds, the number of all alternative mappings is exponential on the number of pairs of schema objects; therefore even reducing it by a fixed percentage may not scale to real-world integration tasks. It is worth noting that in the majority of experimental assessments outlined in the papers mentioned in this survey, the experiments were conducted on data sets with a limited number of schema objects.

The p–datalog programs generated by the schema matching phase described in Nottelmann and Straccia [2005; 2007] are also used to query the mediated schema. For example, consider again the rule:

.8 pubyear(d,y) ← year(d,y).

Now, assume that from an information extraction engine we know that 50% of the occurrences of years in document d1 have the value 2003 (this can be indicated by a rule .5 year(d1,2003)). The authors conclude that the probability of 2003 to be the publication year of document d1 is $.8 \cdot .5 = .4$. This result is obtained using an implicit assumption of probabilistic independence: in particular, $P(\text{pubyear}(d, y)|\text{year}(d, y)) = P(\text{pubyear}(d, y))$. Note that also in this case the assumption may be wrong, for example, if all occurrences of the year 2003 represent publication years, leading to a probability of .5. However, this work as well as the approach presented in Calì and Lukasiewicz [2006; 2008] and Calì et al. [2008] are certainly valid starting points for analyzing query answering, because they are grounded on sound and well studied theories.

In addition to these works, query answering over uncertain data is discussed in the papers mentioned in Section 3.1. In particular, the query language defined in Tseng et al. [1993] has been proposed as a language to query heterogeneous databases. However as these works do not deal with specific aspects of the data integration process, we will not provide additional details here.

3.5.1 *Aggregate Query Answering.* The aforementioned by-table and by-tuple semantics can also be used for computing aggregate queries. Gal et al. [2009] present three possible kinds of results for each of the basic aggregate operators COUNT, MIN, MAX, SUM, and AVG: (1) the *distribution* of all the possible answers, annotated with their probability, (2) an interval bounding the *range* of the possible answers and (3) the *expected value* of the query and here we notice that *range* and *expected value* can be computed from the complete *distribution*—consequently more expressive. The authors present PTIME algorithms for all the operators under the by-table semantics, and also under the by-tuple semantics for the COUNT operator, the SUM operator (limited to *range* and *expected value* results) and the MAX, MIN, and AVG operators (limited to *range* results).

3.5.2 *Top-k Query Answering.* One of the most usual ways to query uncertain data is to return the k best answers, where *best* usually means *most likely* in probabilistic frameworks. Dong et al. [2007] provide a greedy algorithm

to compute top-k answers: the main idea behind it is that some data sources with a small probability of matching may not change the list of top-k results of high probability. Therefore, the answers are computed starting from high-probability correspondences, and if we notice the probability of the remaining ones is not sufficient to change the list of top-k answers, we stop the computation. In the worst case, the algorithm must compute all the answers and therefore an experimental evaluation will be necessary to reach a better understanding of the practical impact of this approach. Top-k schema matching is included in the tool demonstrated in Roitman et al. [2008], although without focusing on query answering and some experimental results are reported in Magnani and Montesi [2009a].

As shown in this section, query answering is a complex topic, and presents a number of open problems.

(1) The complexity of query evaluation has been studied only with regard to specific classes of queries, for example, select-project-join, and data models.
(2) Efficient query processing depends on the presence of a small number of alternative mappings; therefore methods to reduce them should be developed. The approaches reviewed in this survey provide only a partial treatment of this problem.
(3) When a user executes a query and receives a result, some very useful feedback can be generated which the system can then use to improve the mappings. Also in this case, a number of solutions have been proposed, but the problem has not been addressed systematically.

## 4. OPEN PROBLEMS

This section summarizes the contributions and open problems that have emerged in this survey on uncertainty management in data integration.

### 4.1 Matching Uncertain Data

So far, this topic has only been studied in Agrawal et al. [2008], but it is also mentioned in some other works as one of the most relevant future research directions [Magnani and Montesi 2007; Dong et al. 2007]. In addition to its theoretical interest, the ability to match uncertain data is necessary for defining a closed integration process and enabling the definition of an algebra of model management operators [Bernstein 2003] and thus the iterative execution of data integration activities on several local data sources.

### 4.2 Generation of Uncertainty Measures

The papers we include in this survey define many matchers producing uncertainty measures, as indicated in Table III. However, in probabilistic approaches each matcher should return probabilities that can be compared and aggregated with the ones produced by other matchers. It should therefore be clear which interpretation of the theory is used inside each matcher, for example, classical, frequentistic, or subjective, and also whether different probability distributions can be aggregated as they are. For example, assume a matcher finds

some common instances inside two schema objects, supporting a *match* relationship, and another matcher thinks that the names of the two objects are not related, supporting a *not match*. How much probability should we assign to the two hypotheses? Surely we can tune these values after some experiments. However this is quite different from having a sound underlying interpretation of the numbers produced by the system, which by the way is the main reason for using probability theory. At the moment, probabilities are almost always generated using ad hoc methods.

Although it does not seem possible to solve this problem in general, we can still study and address it in specific domains, such as in the context of by-tuple semantics. These may have a frequentistic interpretation and allow for experimental evaluations of the impact of different choices on the results of the integration process.

### 4.3 Evolution of Matcher Ensembles

In the reviewed works, the pools of matchers used to analyze the data are static: they do their job independently of each other, and their outcomes are aggregated. In our opinion, one of the future trends will be to study how the quality of the identified relationships may be affected by more dynamic sets of matchers. For instance, as de Keijzer and van Keulen [2007] have suggested, user feedback may be used to change the knowledge base, in the same way we may expect that it could be used to update the behavior of some matchers—by either punishing or reinforcing them.

### 4.4 Aggregation of Matcher Outcomes

Some matchers may be mutually independent meaning that we can change the features analyzed by one of them without affecting the outcome of the other. As an example, consider two matchers that respectively compare the number of distinct values in two columns of a relation and their names. If we change the names, the outcome of the cardinality matcher will not change, and vice versa. However, in general different matchers can be interdependent. As a consequence, a general method that performs the combination needs to know the single uncertain relationships produced by different matchers *and* additional information about their probabilistic dependencies.

The system described in Magnani and Montesi [2009a] uses two kinds of matchers: one to analyze the data, and another to aggregate the results of correlated matchers. However, only some simple ad hoc matchers have been implemented without a general theory. Other papers using probabilistic matchers use independence assumptions or allow the impact of the matchers to be weighed as in Nottelmann and Straccia [2005]. However no one has as yet proposed general solutions to this problem.

A different approach, followed in some of the reviewed papers, is to try to mimic human behavior instead of sticking to the axioms of probability that may not effectively describe how a human expert would perform a manual integration activity. This approach has been adopted in Magnani and Montesi [2008b], where probabilities were replaced by preferences. In that work, the

authors present a parametric model in which many alternative aggregation operators can be used. However, this approach runs the risk of defining a number of ad hoc methods without evidence of their effectiveness. This leads to the open problem of evaluating these approaches which shall be discussed separately.

### 4.5 Generation of Mapping

Another aggregation of probabilities occurs when we generate an uncertain mapping from a set of uncertain relationships; in this case too there can be probabilistic dependencies. Similarly, other dependencies not known a priori can derive from the semantics of the local data sources.

Carefully selected independence assumptions and approximated or nonprobabilistic methods should be used to tackle this problem which is currently still awaiting a satisfactory answer.

### 4.6 Generation of Mediated Schemata

When enabling a fully automated data integration process, we cannot assume the mediated schema is already available. Therefore we need methods to learn it from the data sources. Several of the surveyed papers have considered this problem but it appears to be an emergent one in this literature.

### 4.7 Query Answering and Complexity Reduction

As we have mentioned above, query answering is tractable in presence of a limited number of possible mappings. Therefore techniques to reduce them are the key to enable the development of real systems. Unfortunately, this corresponds to the removal of possibly correct mappings and brings us back to traditional methods. It follows that the usage of thresholds or top-k algorithms, with a small k, should be performed with care. In addition, we have seen that when defining a top-k algorithm we should know the probability of the alternative mappings and this is unfeasible in general cases with probabilistic dependencies.

### 4.8 User Involvement

So far, it is unclear how the uncertain results of a data integration process can be used effectively. Traditional methods use top-k results to include the user into the matching phase and end with one correct mapping. Some recent works suggest generating a single certain schema, called *consolidated schema* so as to summarize an uncertain mediated schema [Sarma et al. 2008c; Magnani and Montesi 2009b]. In general, it is of primary importance to develop adequate user interfaces for accessing these complex results. Otherwise, in many cases the result of an uncertain integration process would not be of any practical use. Similarly, it is an open question how to use an uncertain mediated schema in a completely automated scenario.

### 4.9 Evaluation

The works considered in this survey cover many complementary aspects of uncertainty management in data integration. The use of our common framework has enabled us to highlight their relationships. However, additional efforts are required in order to evaluate and compare these techniques such as, for example, developing benchmark data. Moreover case studies that report on the application of these methods would be useful in identifying more practical problems.

## 5. CONCLUSION

This article describes the status of research on uncertainty management in data integration. It provides a snapshot of a rapidly evolving area. Uncertain data integration systems have many potential applications such as Web 2.0 online repositories updated by millions of users that automatically generate their mediated schemata, or online and automatically updated interfaces to scientific data sources. Applications of uncertain data integration have a vast number of potential users and therefore we expect them to play an important role in popularizing next-generation information systems supporting uncertainty.

The advantages of using uncertain data models to represent the result of data integration tasks, that is, models that represent both data and their degree of certainty, have not been studied thoroughly and are not easy to assess experimentally. Data integration methods that manage uncertainty are certainly closer to typical human behavior than traditional methods; they are interesting from an academic point of view as alternative approaches for investigation and are intuitively reasonable. However, they remain as yet untested in real-world scenarios. Some preliminary experiments potentially supporting their effectiveness were presented in Gal [2006a] and Magnani and Montesi [2007]. These results were obtained by studying the distribution of correct relationships inside top-k mappings. The analysis of precision and the recall of correct relationships shows that it is often worth keeping uncertainty after the matching phase as many correct relationships are absent in top-1 (the most likely) mapping. In addition, these experiments show that in complex data integration tasks, the information loss caused by the removal of uncertainty may be relevant. Indeed, even if we consider k alternative mappings we risk losing some correct relationships unsupported by high degrees of confidence.

The majority of methods presented are quantitative and suffer from complexity problems. In fact, although it has been shown that many classes of queries can be answered in polynomial time with respect to data size and mapping, mapping size can be exponential in terms of the size of the input schemata. Qualitative approaches have been presented claiming reduced complexity but the impact on the precision and recall of the methods with respect to probabilistic systems has not been studied yet.

The main problem related to probabilistic methods lies in the impact of probabilistic dependencies which may affect many of the steps in the process and are often influenced by the adopted data models. For example, Magnani et al. [2005] show that relationships connecting common entities in an ER diagram

are a source of dependencies that increase the complexity of the matching and merging steps. To avoid this problem, the use of less expressive schema languages (i.e., sets of schema objects without connections) and relationships ($M$ and $\not M$) have been proposed but it is still unclear if these are good compromises, that is, if they can be used effectively in real applications.

Moreover, it is worth noticing that many open problems are more related to the adopted uncertainty management formalisms than to the specific application context of data integration. For example, the computational problems of the reviewed probabilistic systems are the same ones that affect other application areas of probability theory. Therefore, to find answers to the open problems discussed in the previous section, it is fundamental to make advances in related fields ranging from the interpretation of mathematical theories of uncertainty to modeling, implementing, and visualizing uncertain data.

A final consideration concerns the development of uncertain data integration systems. These systems obviously present specific features like enhanced user interfaces and uncertainty manipulation modules. However, the architecture of uncertain and traditional data integration systems is basically the same. In addition, the matchers (the most important components of these systems) may adopt all the techniques that have already been developed and which are available in traditional systems, with minor adjustments to make them produce uncertain results expressed using the preferred uncertainty management theory. It follows, therefore, that the new capabilities described in this article are likely to be adopted gracefully without the need to develop new systems from scratch, and this will certainly facilitate their future development.

## REFERENCES


AGRAWAL, P., BENJELLOUN, O., SARMA, A. D., HAYWORTH, C., NABAR, S. U., SUGIHARA, T., AND WIDOM, J. 2006. Trio: A system for data, uncertainty, and lineage. In *Proceedings of the 32nd International Conference on Very Large Data Bases*. ACM, 1151–1154.

AGRAWAL, P., SARMA, A. D., ULLMAN, J., AND WIDOM, J. 2008. Lav integration of uncertain data. Tech. rep. 2008-31, Stanford InfoLab.

AL-KHALIFA, S., YU, C., AND JAGADISH, H. V. 2003. Querying structured text in an XML database. In *Proceedings of the SIGMOD Conference*.

ALTAREVA, E. AND CONRAD, S. 2001. The problem of uncertainty and database integration. In *Proceedings of the Workshop on Engineering Federated Information Systems (EFIS)*. 92–99.

ALTAREVA, E. AND CONRAD, S. 2003. Statistical analysis as methodological framework for data(base) integration. Lecture Notes in Computer Science, vol. 2813. Springer, 17–30.

ALTAREVA, E. AND CONRAD, S. 2005. Evaluating and improving integration quality for heterogeneous data sources using statistical analysis. In *Proceedings of the International Database Engineering and Applications Symposium*. IEEE Computer Society, 406–414.

BARBARA, D., GARCIA-MOLINA, H., AND PORTER, D. 1992. The management of probabilistic data. *IEEE Trans. Knowl. Data Engin. 4*, 5, 487–501.

BATINI, C. AND SCANNAPIECO, M. 2006. *Data Quality: Concepts, Methodologies, and Techniques*. Data-Centric Systems and Applications. Springer.

BENFERHAT, S., DUBOIS, D., KACI, S., AND PRADE, H. 2006. Bipolar possibility theory in preference modeling: Representation, fusion, and optimal solutions. *Inform. Fusion 7*, 1, 135–150.

BERNSTEIN, P. A. 2003. Applying model management to classical meta data problems. In *Proceedings of the Conference on Innovative Data Systems Research (CIDR)*.

BONISSONE, P. P. AND TONG, R. M. 1985. Editorial: Reasoning with uncertainty in expert systems. *Int. J. Man–Mach. Stud. 22*, 3, 241–250.

BOSC, P. AND PRADE, H. 1996. An introduction to the fuzzy set and possibility theory-based treatment of flexible queries and uncertain or imprecise databases. In *Uncertainty Management in Information Systems*. 285–324.

BOULOS, J., DALVI, N. N., MANDHANI, B., MATHUR, S., RÉ, C., AND SUCIU, D. 2005. Mystiq: A system for finding more answers by using probabilities. In *Proceedings of the SIGMOD Conference*. 891–893.

CALÌ, A. AND LUKASIEWICZ, T. 2006. An approach to probabilistic data integration for the semantic Web. In *Proceedings of the ISWC-URSW Conference*.

CALÌ, A. AND LUKASIEWICZ, T. 2008. An approach to probabilistic data integration for the semantic Web. In *Proceedings of the ISWC International Workshops Uncertainty Reasoning for the Semantic Web I*. Lecture Notes in Computer Science, vol. 5327. Springer, 52–65.

CALÌ, A., LUKASIEWICZ, T., PREDOIU, L., AND STUCKENSCHMIDT, H. 2008. Rule-based approaches for representing probabilistic ontology mappings. In *Proceedings of the ISWC International Workshops Uncertainty Reasoning for the Semantic Web I*. Lecture Notes in Computer Science, vol. 5327. Springer, 66–87.

CHENG, R., SINGH, S., AND PRABHAKAR, S. 2005. U-dbms: A database system for managing constantly-evolving data. In *Proceedings of the 31st International Conference on Very Large Data Bases*. ACM, 1271–1274.

CODD, E. F. 1979. Extending the database relational model to capture more meaning. *ACM Trans. Datab. Syst. 4*, 4, 397–434.

DE KEIJZER, A. AND VAN KEULEN, M. 2007. User feedback in probabilistic integration. In *Proceedings of the 18th International Conference on Database and Expert Systems Applications (DEXA'07)*. IEEE Computer Society, 377–381.

DE KEIJZER, A. AND VAN KEULEN, M. 2008. Imprecise: Good-is-good-enough data integration. In *Proceedings of the 24th International Conference on Data Engineering*. IEEE Computer Society Press, Los Alamitos, CA.

DE KEIJZER, A., VAN KEULEN, M., AND LI, Y. 2006. Taming data explosion in probabilistic information integration. In *Proceedings of the International Workshop on Inconcistency and Incompleteness in Databases*.

DEKHTYAR, A., GOLDSMITH, J., AND HAWKES, S. R. 2001. Semistructured probalistic databases. In *Statistical and Scientific Database Management*.

DEMOLOMBE, R. 1997. Uncertainty in intelligent databases. In *Uncertainty Management in Information Systems*, A. Motro and C. Thanos, Eds. Kluwer.

DEY, D. AND SARKAR, S. 1996. A probabilistic relational model and algebra. *ACM Trans. Datab. Syst. 21*, 3, 339–369.

DEY, D., SARKAR, S., AND DE, P. 2002. A distance-based approach to entity reconciliation in heterogeneous databases. *IEEE Trans. Knowl. Data Engin. 14*, 3, 567–582.

DO, H.-H. AND RAHM, E. 2007. Matching large schemas: Approaches and evaluation. *Inform. Syst. 32*, 6.

DOAN, A. AND HALEVY, A. Y. 2005. Semantic integration research in the database community: A brief survey. *AI Mag. 26*, 1, 83–94.

DONG, X. L., HALEVY, A. Y., AND YU, C. 2007. Data integration with uncertainty. In *Proceedings of the 33rd International Conference on Very Large Data Bases*. ACM, 687–698.

EUZENAT, J. AND SHVAIKO, P. 2007. *Ontology matching*. Springer-Verlag, Berlin.

FLORESCU, D., KOLLER, D., AND LEVY, A. Y. 1997. Using probabilistic information in data integration. In *Proceedings of the International Conference on Very Large Data Bases*. 216–225.

FUHR, N. 1995. Probabilistic datalog - a logic for powerful retrieval methods. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM Press, 282–290.

FUHR, N. AND RÖLLEKE, T. 1997. A probabilistic relational algebra for the integration of information retrieval and database systems. *ACM Trans. Inform. Syst. 15*, 1, 32–66.

GAL, A. 2006a. Managing uncertainty in schema matching with top-k schema mappings. *J. Data Seman. VI 4090*, 90–114.

GAL, A. 2006b. Why is schema matching tough and what can we do about it? *SIGMOD Record 35*, 4, 2–5.

GAL, A. 2008. Interpreting similarity measures: Bridging the gap between schema matching and data integration. In *Proceedings of the 24th International Conference on Data Engineering Workshops*. IEEE Computer Society, 278–285.

GAL, A., ANABY-TAVOR, A., TROMBETTA, A., AND MONTESI, D. 2005. A framework for modeling and evaluating automatic semantic reconciliation. *VLDB J. 14*, 1, 50–67.

GAL, A., MARTINEZ, M. V., SIMARI, G. I., AND SUBRAHMANIAN, V. 2009. Aggregate query answering under uncertain schema mappings. In *Proceedings of the 25th International Conference on Data Engineering*.

HALEVY, A. Y. 2003. Data integration: A status report. In *Proceedings of the Conference on Business, Technology, and the Web (BTW)*. Lecture Notes in Informatics, vol. 26. GI, 24–29.

HALEVY, A. Y., FRANKLIN, M. J., AND MAIER, D. 2006a. Principles of dataspace systems. In *Proceedings of the 25th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*. 1–9.

HALEVY, A. Y., RAJARAMAN, A., AND ORDILLE, J. J. 2006b. Data integration: The teenage years. In *Proceedings of the International Conference on Very Large Data Bases*. 9–16.

HAYNE, S. AND RAM, S. 1990. Multi-user view integration system (muvis): An expert system for view integration. In *Proceedings of the 6th International Conference on Data Engineering*. IEEE Computer Society, 402–409.

HUNG, E., GETOOR, L., AND SUBRAHMANIAN, V. 2003a. Probabilistic interval XML. In *Proceedings of the International Conference on Database Theory (ICDT)*.

HUNG, E., GETOOR, L., AND SUBRAHMANIAN, V. 2003b. PXML: A probabilistic semistructured data model and algebra. In *Proceedings of the International Conference on Data Engineering*.

HUNTER, A. AND LIU, W. 2006a. Fusion rules for merging uncertain information. *Inform. Fusion. 7*, 1, 97–134.

HUNTER, A. AND LIU, W. 2006b. Merging uncertain information with semantic heterogeneity in XML. *Knowl. Inform. Syst. 9*, 2, 230–258.

KLEMENT, E. P., MESIAR, R., AND PAP, E. 2000. *Triangular Norms*. Kluwer, Dordrecht.

LAKSHMANAN, L. V. S., LEONE, N., ROSS, R., AND SUBRAHMANIAN, V. S. 1997. ProbView: A flexible probabilistic database system. *ACM Trans. Datab. Syst. 22*, 3, 419–469.

LEE, S. K. 1992. An extended relational database model for uncertain and imprecise information. In *Proceedings of the International Conference on Very Large Data Bases*, L.-Y. Yuan, Ed.

LENZERINI, M. 2002. Data integration: A theoretical perspective. In *Proceedings of the PODS Conference*. 233–246.

LOUIE, B., DETWILER, L., DALVI, N. N., SHAKER, R., TARCZY-HORNOCH, P., AND SUCIU, D. 2007. Incorporating uncertainty metrics into a general-purpose data integration system. In *Proceedings of the 19th International Conference on Scientific and Statistical Database Management*. IEEE Computer Society.

MAGNANI, M. AND MONTESI, D. 2007. Uncertainty in data integration: current approaches and open problems. In *Proceedings of the VLDB Workshop on Management of Uncertain Data*.

MAGNANI, M. AND MONTESI, D. 2008a. Management of interval probabilistic data. *Acta Informatica 45*, 2, 93–130.

MAGNANI, M. AND MONTESI, D. 2008b. Preference-based uncertain data integration. In *Proceedings of the 16th International Conference on Knowledge Engineering: Practice and Patterns (EKAW)*. Lecture Notes in Computer Science, vol. 5268. Springer, 136–145.

MAGNANI, M. AND MONTESI, D. 2009a. Probabilistic data integration. Tech. rep. UBLCS-2009-10, University of Bologna.

MAGNANI, M. AND MONTESI, D. 2009b. Towards schema uncertainty. In *Proceedings of the 3rd International Conference on Scalable Uncertainty Management (SUM)*. Springer.

MAGNANI, M., RIZOPOULOS, N., MCBRIEN, P., AND MONTESI, D. 2005. Schema integration based on uncertain semantic mappings. In *Proceedings of the 24th International Conference on Conceptual Modeling*. Lecture Notes in Computer Science, vol. 3716. Springer, 31–46.

MARIE, A. AND GAL, A. 2007. Managing uncertainty in schema matcher ensembles. In *Proceedings of the 1st International Conference on Scalable Uncertainty Management*. Lecture Notes in Computer Science, vol. 4772. Springer, 60–73.

MIMNO, D., MCCALLUM, A., AND MIKLAU, G. 2007. Probabilistic representations for integrating unreliable data sources. In *Proceedings of the Workshop on Information Integration on the Web (IIWeb)*.

MOTRO, A. 1995. Imprecision and uncertainty in database systems. In *Fuzziness in Database Management Systems*, P. Bosc and J. Kacprzyk, Eds. Physica-Verlag, 3–22.

NAGY, M., VARGAS-VERA, M., AND MOTTA, E. 2007. Dssim—managing uncertainty on the semantic Web. In *Proceedings of the International Workshop on Ontology Mapping*.

NIERMAN, A. AND JAGADISH, H. V. 2002. ProTDB: Probabilistic data in XML. In *Proceedings of the VLDB Conference*.

NOTTELMANN, H. AND STRACCIA, U. 2005. splmap: A probabilistic approach to schema matching. In *Proceedings of the European Conference on Information Retrieval*. 81–95.

NOTTELMANN, H. AND STRACCIA, U. 2007. Information retrieval and machine learning for probabilistic schema matching. *Inform. Process. Manage. 43*, 3, 552–576.

OLTEANU, A., MUSTIÈRE, S., AND RUAS, A. 2008. Matching imperfect spatial data. In *Proceedings of the 7th International Symposium on Spatial Accuracy Assessment in Natural Resources and Environmental Sciences*.

PAL, N. R. 1999. On quantification of different facets of uncertainty. *Fuzzy Set Syst. 107*, 81–91.

PANKOWSKI, T. 2008. Reconciling inconsistent data in probabilistic XML data integration. In *Proceedings of the 25th British National Conference on Databases*. Lecture Notes in Computer Science, vol. 5071. Springer, 75–86.

PITTARELLI, M. 1994. An algebra for probabilistic databases. *IEEE Trans. Knowl. Data Engin. 6*, 2, 293–303.

RAHM, E. AND BERNSTEIN, P. A. 2001. A survey of approaches to automatic schema matching. *VLDB J. 10*, 4, 334–350.

RE, C., DALVI, N. N., AND SUCIU, D. 2007. Efficient top-k query evaluation on probabilistic data. In *Proceedings of the 23rd International Conference on Data Engineering*. IEEE, 886–895.

ROITMAN, H., GAL, A., AND DOMSHLAK, C. 2008. Providing top-k alternative schema matchings with ontomatcher. In *Proceedings of the 27th International Conference on Conceptual Modeling*. Lecture Notes in Computer Science, vol. 5231. Springer.

SARMA, A. D., BENJELLOUN, O., HALEVY, A. Y., AND WIDOM, J. 2006. Working models for uncertain data. In *Proceedings of the 22nd International Conference on Data Engineering*. IEEE Computer Society, 7.

SARMA, A. D., DONG, X., AND HALEVY, A. 2008a. Bootstrapping pay-as-you-go data integration systems. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*. ACM, 861–874.

SARMA, A. D., DONG, X., AND HALEVY, A. 2008b. *Managing and Mining Uncertain Data*. Springer, Chapter Uncertainty in data integration.

SARMA, A. D., DONG, X., AND HALEVY, A. Y. 2008c. Bootstrapping pay-as-you-go data integration systems. In *Proceedings of the SIGMOD Conference*. 861–874.

SHAFER, G. 1976. *A Mathematical Theory of Evidence*. Princeton University Press.

SMETS, P. 1997. Imperfect information: Imprecision - uncertainty. In A. Motro and Ph. Smets Eds. *Uncertainty Management in Information Systems, from Needs to Solutions*. Kluwer Academic Publishers, 225–254.

SMITHSON, M. J. 1989. *Ignorance and Uncertainty: Emerging Paradigms*. Springer Verlag.

TSENG, F. S.-C., CHEN, A. L. P., AND YANG, W.-P. 1993. Answering heterogeneous database queries with degrees of uncertainty. *Distrib. Paral. Datab. 1*, 3, 281–302.

VAN KEULEN, M., DE KEIJZER, A., AND ALINK, W. 2005. A probabilistic XML approach to data integration. In *Proceedings of the 21st International Conference on Data Engineering*. 459–470.

WANG, Y., LIU, W., AND BELL, D. A. 2007. Combining uncertain outputs from multiple ontology matchers. In *Proceedings of the 1st International Conference on Scalable Uncertainty Management*. Lecture Notes in Computer Science, vol. 4772. Springer, 201–214.

WIDOM, J. 2005. Trio: A system for integrated management of data, accuracy, and lineage. In *Proceedings of the Conference on Innovative Data Systems Research (CIDR)*. 262–276.

WITOLD LIPSKI, J. 1979. On semantic issues connected with incomplete information databases. *ACM Trans. Datab. Syst. 4*, 3, 262–296.

WORBOYS, M. F. AND CLEMENTINI, E. 2001. Integration of imperfect spatial information. *J. Vis. Lang. Comput. 12*, 1, 61–80.

Received February 2008; revised August 2008, March 2009, July 2009; accepted November 2009