2000

Imprecise Answers in Distributed Environments: Estimation of Information Loss for Multi-Ontology Based Query Processing

Eduardo Mena

Vipul Kashyap

Arantza Illarramendi

Amit P. Sheth
Wright State University - Main Campus, amit@sc.edu

IMPRECISE ANSWERS IN DISTRIBUTED ENVIRONMENTS: ESTIMATION OF INFORMATION LOSS FOR MULTI-ONTOLOGY BASED QUERY PROCESSING

E. MENA
IIS department, University of Zaragoza, María de Luna 3
50015 Zaragoza, Spain

and

V. KASHYAP
MCC-1G332R, Bellcore, 445 South St.
Morristown, NJ 07960, USA

and

A. ILLARRAMENDI
LSI department, University of the Basque Country, Apdo. 649
20080 San Sebastián, Spain

and

A. SHETH
LSDIS Lab., University of Georgia
Athens, GA 30602, USA

Recieved November 30, 1998
Revised October 28, 1999

ABSTRACT

The World Wide Web is fast becoming a ubiquitous computing environment. Prevalent keyword-based search techniques are scalable, but are incapable of accessing information based on concepts. We investigate the use of concepts from multiple, real-world pre-existing, domain ontologies to describe the underlying data content and support information access at a higher level of abstraction. It is not practical to have a single domain ontology to describe the vast amounts of data on the Web. In fact, we expect multiple ontologies to be used as different world views and present an approach to "browse" ontologies as a paradigm for information access. A critical challenge in this approach is the vocabulary heterogeneity problem. Queries are rewritten using interontology relationships to obtain translations across ontologies. However, some translations may not be semantics preserving, leading to uncertainty or loss in the information retrieved. We present a novel approach for estimating loss of information based on navigation of ontological terms. We define measures for loss of information based on intensional information as well as on well established metrics like *precision* and *recall* based on extensional information. These measures are used to select results having the desired quality of information.

1. Introduction

The World Wide Web has fast become a preferred information access and application support environment for a large number of computer users. In most cases, there is no centralized or controlled information management, as anyone can make a wide variety of data available on the Web. This has facilitated an exponential growth in the accessible information on the Web. In distributed and federated database systems [23], logical integration of the schemas describing the underlying data is used to handle the structural and representational heterogeneity. In a tightly coupled federated database approach, the relationships are fixed at schema integration or definition time. In a loosely coupled federated database (or multidatabase) approach, the relationships are defined when defining the multidatabase view prior to querying the databases. Neither of these options are feasible nor appealing in the much more diversified and unmanaged Web-centric environment.

Use of domain specific ontologies is an appealing approach to allow users to express information requests at a higher level of abstraction compared to only keyword based access. As discussed in [12], we view ontologies as the specification of a representational vocabulary for a shared domain of discourse which may include definitions of classes, relations, functions and other objects. Since one cannot expect a single ontology to describe the vast amounts of data on the Web, we believe it is necessary the use of multiple domain specific ontologies as different world views describing the wide variety of data and capturing the needs of a varied community of users. A critical issue that prevents wide spread use of ontologies is the labor intensive nature of the process of designing and constructing an ontological specification. In the OBSERVER[1] system, we demonstrate our approach of using multiple pre-existing real-world domain ontologies to access heterogeneous, distributed and independently developed data repositories. This enables the use of "off the shelf ontologies", thus minimizing the need of designing ontologies from scratch.

One consequence of our emphasis on ontology re-use is that they are developed independently of the data repositories and have been used to describe information content in data repositories independently of the underlying syntactic representation of the data [14]. New repositories can be easily added to the system by mapping ontological concepts to data structures in those repositories. This approach is more suitable for open and dynamic environments such as the Web, and allows each data repository to be viewed at the level of the relevant semantic concepts.

We present an approach for browsing multiple related ontologies for information access. A user query formulated using terms in some user view/domain ontology is translated by using terms of other (target) domain ontologies. Mechanisms dealing with incremental enrichment of the answers are used. The substitution of a term by traversing interontological relationships like *synonyms* (or combinations of them [17, 18]) and combinations of *hyponyms* (specializations) and *hypernyms* (generalizations) provides answers not otherwise available by using only a single ontology. However, this usually changes the semantics of the query. The main contribution of this paper is the use of mechanisms to estimate loss of information (based on intensional and extensional properties) in the face of possible semantic changes when

[1] *Ontology Based System Enhanced with Relationships for Vocabulary hEterogeneity Resolution.*

translating a query across different ontologies. It may be noted that in our approach thousands of data repositories may be described by tens of ontologies. In general, a user may be willing to sacrifice the quality of information for a quicker response from the system, as discussed in [24].

Several projects that deal with the problem of interoperable systems can be found in the literature, e.g., TSIMMIS [6], SIMS [1], Information Manifold [15], InfoSleuth [2], Carnot [8], etc. The commonalities between their approaches and ours can be summed up as: (a) some way of using a high level semantic view (ontology) to describe data content; and (b) use of specialized wrappers to access underlying data repositories. In this paper, however, we present novel techniques to estimate the loss of information incurred when translating user queries into other ontologies. This measure of loss (whose upper limit is defined by the user) guides the system in navigating those ontologies that have more relevant information; it also provides the user with a "level of confidence" in the answer that may be retrieved. We use well-established metrics, like precision and recall, and adapt them to our context in order to measure the change in semantics incurred when providing an answer with a certain degree of imprecision. This approach contracts with the measure of the change in the extension used in classical Information Retrieval [22].

There have been approaches in the research literature for approximate query answering in situations where answers are obtained from multiple information sources. Most approaches attempt to estimate some measure of divergence from the true answer and are based on some technique of modeling uncertainty. In the Multiplex project [20], the soundness and completeness of the results are estimated based on the intersections and unions of the candidate results. In our approach, the Information Retrieval analogs of soundness (precision) and completeness (recall) are estimated based on the sizes of the extensions of the terms. We combine these two measures to compute a composite measure using a numerical value. This can then be used to choose the answers with the least loss of information. Alternatively, numerical probabilistic (possibilistic) measures are used in [25, 10], but are based on *ad hoc* estimates of the initial probability (possibility) values. In our approach we provide a set theoretic basis for the estimation of information loss measures.

The rest of the paper is organized as follows[2]. Section 2 introduces the query processing strategy in OBSERVER and briefly discusses the translation mechanisms when synonym, hyponym and hypernym relationships are used for controlled and incremental query expansion to different target ontologies. Section 3 describes the techniques used to estimate imprecision in information retrieval using intensional and extensional information. Section 4 justifies the approach used to measure the information loss. In Section 5 we discuss the estimation of the loss across correlated answers. Finally, conclusions are presented in Section 6.

2. Query Processing in OBSERVER

The idea underlying our query processing algorithm is the following: give the first possible answer and then enrich it in successive iterations until the user is satisfied.

[2]To avoid duplication and for brevity, we do not repeat much of the basic discussion on query processing approach and prototype system architecture which appears in [17] and focus here only on the new contributions.

Notice that thousands of data repositories described by tens of ontologies could be available in our context, so users will prefer to get a good set of semantically correct data rather than waiting for a long time until all the relevant data in the Global Information System have been retrieved. Moreover, a certain degree of imprecision (defined by each user) in the answer could be allowed if it helps to speed up the search of the requested information.

In the following we first present the main steps of our query processing approach, then we show some features related to the Description Logics (DL) expressions that comprise queries, and we finish with an illustrative example.

2.1. Query Processing: Main Steps

The three main steps of the proposed query processing approach are the following:

Step 1: Query Formulation based on the user ontology

The user browses the available ontologies (which are ordered by knowledge areas) and chooses a *user ontology* that contains the terms needed to express the semantics of her/his information needs. Then, with the help of a GUI, the user chooses terms from the user ontology to build the constraints and projections[3] that comprise the query.

Step 2: Access data underlying the user ontology and present the answer

The DL expression that comprises the query is translated, with the help of *mappings*[4] [3] of the terms involved in such an expression, into several subqueries expressed in the local query language of the underlying repositories[5]. To perform this task the system uses different translators and wrappers. Moreover, answers coming from different data sources must be translated into the "language" of the user ontology to facilitate removal of redundant objects and to update incomplete objects. Thus, the answer is correlated and presented to the user. A more detailed description of this step appears in [11, 17].

Step 3: Controlled and Incremental Query Expansion to Multiple Ontologies

If the user is not satisfied with the answer, the query processor retrieves more data from other ontologies in the Global Information System to "enrich" the answer in an incremental manner. Some researchers have looked into the problem of query relaxation [7, 5]. However, they have proposed techniques for query relaxation within the same schema/ontology/knowledge base. We differ with the above in two important ways: (1) we propose techniques for query relaxation *across* ontologies by using synonym, hyponym and hypernym relationships; and (2) we provide techniques to estimate the loss of information incurred.

[3] These projections and constraints are internally expressed by the system as an expression in Description Logics (DL) [9] that represents the user query.

[4] Mappings in our approach are expressions of Extended Relational Algebra that relate terms in ontologies with the underlying data elements.

[5] In the case of relational databases, the DL expression is translated into a list of SQL sentences.

In our system, a new component ontology, which we call the *target ontology*, is selected from the Global Information System. The user query must be expressed/translated using terms of that target ontology. For that task, the user and target ontologies are integrated (see [18]) by using the interontology relationships defined between them. When a new ontology is made available to the Global Information System, the semantic relationships between its terms and other terms in other ontologies must be defined in a module called the *Interontology Relationship Manager (IRM)* [17]. The IRM is the key for managing different component ontologies without missing the semantics of each one. Thus, this module manages the *semantic properties* between terms in different ontologies, concretely it deals with synonyms, hyponyms and hypernyms. This information allows integration of two given ontologies in the system without user intervention. Other authors [13, 4] have suggested different sets of relationships.

When the user and target ontology are integrated automatically by the system, the user query is rewritten and classified in the integrated ontology. Two situations can occur during this process:

(i) All the terms in the user query may have been rewritten by their corresponding synonyms in the target ontology. Thus the system obtains a semantically equivalent query (*full translation*) and no loss of information is incurred. In this case, the plan to obtain the answer consists of accessing the data corresponding to the translated query expression.

(ii) There are terms in the user query that cannot be translated into the target ontology - they do not have synonyms in the target ontology (we call them *conflicting terms*). This kind of translation is called a *partial translation*.

If, after the process of integration, the Query Processor has obtained a partial translation, it tries to combine it with previously obtained partial translations in order to obtain a new full translation [19]. Another alternative is the following: if the user allows the system to provide answers with a certain degree of imprecision, new plans could be generated by substituting the conflicting terms by semantically similar expressions that could lead to a full translation of the user query. So, each conflicting term in the user query is replaced by the intersection of its immediate parents (*hypernyms*) or by the union of its immediate children (*hyponyms*), recursively, until a translation of the conflicting term is obtained using only the terms of the target ontology. Each substitution of a term in the original query implies a certain loss of information.

This process can generate *several* possible translations of the user query into a given target ontology. All the possibilities are explored and the result is a list of plans for which the system will estimate the associated loss.

2.2. Description Logics Expressions

Ontologies are defined using a knowledge representation system based on Description Logics (DL system) [9]. The core of those systems is their concept language, which can be viewed as a set of constructors for denoting concepts and relationships among concepts (roles). Besides concept, role, and individual names, the alphabet

of concept languages includes a number of constructors that permit the formation of concept expressions[6].

The set of constructors for concept expressions considered in this work are presented in Table 1, where 'A' is a concept name, 'B' and 'C' are concept expressions, 'R' is a role name, 'n' is a number and 'i' is an individual.

Constructor name	Syntax used
concept name	A
conjunction	(**AND** B C)
universal quantification	(**ALL** R B)
number restrictions	(**ATLEAST** n R) (**ATMOST** n R)
role fillers	(**FILLS** R i)

Table 1: DL constructors considered and their syntax

As discussed earlier (step 3 of query processing), each conflicting term in the user query is replaced by the intersection of its immediate parents or by the union of its immediate children. This is valid for concept expressions which are simply term names or for concept expressions where only one term appears ('FILLS', 'ATLEAST' and 'ATMOST'). For the case of concept expressions involving two or more terms ('AND' and 'ALL'), we proceed in the following way.

Assume that the concept A subsumes concepts B and C and is subsumed by concepts D and E in the integrated ontology (D and E are the immediate subsumers of A, and B and C are the immediate subsumees of A); see Figure 1.

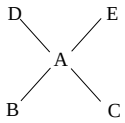

Figure 1: Example for the translation of different concept expressions

AND constructor. Consider the concept expression '*(***AND** *A F)*'. **AND** denotes conjunction and is interpreted as set intersection. The two cases explored in the translation process are:

- A is replaced by the intersection of its immediate parents, $D \cap E$. The concept expression can thus be translated as:

  *(***AND** *D E F)*

- A is replaced by the union of its immediate children, $B \cup C$. Taking into account that we do not consider the 'OR' constructor in DL expressions, we generate the following plan using the union set operation:

  *(***AND** *UNION(B,C) F)*

[6] As far as role expressions are concerned, we only consider role names.

The UNION operation is performed by the OBSERVER system. In DL systems with the 'OR' constructor, the previous substitution would result in a new DL expression, '*(***AND** *(***OR** *B C) F)*'.

ALL constructor. Consider the concept expression (**ALL** R A). **ALL** denotes the set of individuals whose role fillers for 'R are individuals of 'A'. Therefore, the two cases are:

- A is replaced by $D \cap E$. The concept expression can thus be translated as:

  *(***ALL** *R (***AND** *D E))*

- A is replaced by $B \cup C$. Taking into account that we do not consider the 'OR' constructor in DL expressions, we generate the following plan using the union set operation:

  *(***ALL** *R UNION(D,E))*

  If the DL system allows the 'OR' operator, the resulting DL expression would be '*(***ALL** *R (***OR** *D E))*'.

Moreover, we assume that roles are translated by appropriate synonyms in the target ontology. If not, we just consider that the whole concept expression cannot be translated and it will be removed from the translation. The estimation of the loss takes into account this removal.

2.3. Example: Generation of Plans

We now illustrate the computation of the plans obtained by processing the following sample query:

$\mathcal{Q}$ = *[NAME PAGES] for (***AND** *BOOK (***FILLS** *CREATOR "Carl Sagan"))*

Suppose now that this query, formulated using terms of some user ontology, is translated into another (target) ontology as follows[7]:

$\mathcal{Q}$ = *[title number-of-pages] for (***AND** *BOOK (***FILLS** *doc-author-name "Carl Sagan"))*

The only conflicting term in the translated query is 'BOOK' (it has no translation into terms of the target ontology). Now suppose that the process of obtaining the various plans corresponding to the different translations of the term 'BOOK' (that is not described here due to space limitation) results in the four following:

Plan 1: (**AND** document (**FILLS** doc-author-name "Carl Sagan"))
Plan 2: (**AND** periodical-publication (**FILLS** doc-author-name "Carl Sagan"))
Plan 3: (**AND** journal (**FILLS** doc-author-name "Carl Sagan"))

[7] Terms from the user ontology are in uppercase and terms from the target ontology are in lowercase.

Plan 4: (**AND** UNION(book, proceedings, thesis, misc-publication, technical-report) (**FILLS** doc-author-name "Carl Sagan"))

Notice that 'BOOK' has been translated by the expressions 'document', 'periodical-publication', 'journal' and 'UNION(book, proceedings, thesis, misc-publication, technical-report)', respectively. Details of this translation process can be found in [18].

In order to know which plan is semantically closer to the original user query, the loss of information incurred in each case should be estimated.

3. Estimating the Loss of Information

We present two approaches to measure the change in semantics when a term in a query is replaced by an expression from another ontology (in an attempt to get a full translation of the user query). The first approach is based on intensional information and the second one is based on extensional information.

The change in semantics must be measured not only to allow the system to decide which substitution minimizes the loss of information but also to present to the user some kind of "level of confidence" in the answer.

3.1. Loss of Information Measurement Based on Intensional Information

In our context, and due to the use of DL systems for describing and querying the ontologies, loss of information can be expressed as the terminological difference between two expressions, the user query and its translation. The terminological difference between two expressions consists of those concepts of the first expression that are not subsumed by the second expression. The DL system is able to calculate the difference automatically[8]. Let us show an example:

Original query: *Q = [NAME PAGES] for (***AND** *BOOK (***FILLS** *CREATOR "Carl Sagan"))*
Plan 1: *Q = [title number-of-pages] for (***AND** *document (***FILLS** *doc-author-name "Carl Sagan"))*

Taking into account the following term definitions[9]:

*BOOK = (***AND** *PUBLICATION (***ATLEAST** *1 ISBN)),*
*PUBLICATION = (***AND** *document (***ATLEAST** *1 PLACE-OF-PUBLICATION))*

The terminological difference is, in this case, the concept expressions of $\mathcal{Q}$ not considered in the plan, i.e., '(**AND** (**ATLEAST** 1 ISBN) (**ATLEAST** 1 PLACE-OF-PUBLICATION))'. Therefore, the loss of information based on intensional information corresponding to Plan 1 is "Instead of books written by Carl Sagan, OBSERVER is providing all the documents (even if they do not have an ISBN and place of publication)[10] written by Carl Sagan.". Other examples can be found in [16].

[8] If the specific DL system used lacks of that feature the terminological difference could be calculated anyway with the help of its *subsumption mechanism* (see [9]).

[9] The terminological difference is calculated between the extended definitions.

[10] '(ATLEAST 1 ISBN)' and '(ATLEAST 1 PLACE-OF-PUBLICATION)' are the concept expressions not translated.

A special problem arises when computing intensional loss due to the vocabulary differences. As the intensional loss is expressed using terms of two different ontologies, the explanation might make no sense to the user as it mixes two "vocabularies". The problem can be even worse if both ontologies are expressed in different natural languages. So, the intensional loss can help to understand the loss only in some cases.

In addition to the vocabulary problem, an intensional measure of the loss of information can make it hard for the system to decide between two alternatives, in order to execute first the plan with less loss. Thus, some numeric way of measuring the loss should be explored.

3.2. Loss of Information Measurement Based on Extensional Information

Precision and Recall have been widely used in Information Retrieval literature to measure loss of information incurred when the answer to a query issued to the information retrieval system contains some proportion of *irrelevant* data [22]. These measures have been adapted to our context in the following manner:

Precision = *proportion of retrieved objects that are relevant* = $\frac{|Ext(C-Term) \cap Ext(Expression)|}{|Ext(Expression)|}$

Recall = *proportion of relevant objects that are retrieved* = $\frac{|Ext(C-Term) \cap Ext(Expression)|}{|Ext(C-Term)|}$

where

C-Term = conflicting term to be translated into the target ontology
Ext(C-Term) = extension underlying C-Term = relevant objects[11] (RelevantSet)
Expression = translation of the term, probably incurring in a loss of information
Ext(Expression) = extension underlying Expression = retrieved objects (RetrievedSet)

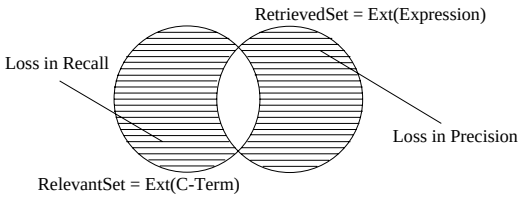

Figure 2: The mismatch between the RetrievedSet and Relevant Set

An expression is, in general, a combination of unions and intersections of terms in the target ontology. Therefore, since at each step the system substitutes conflicting terms by the intersection of its parents or by the union of its children, the estimate size of the extension is an interval with an upper (|Ext(Expr)|.high) and lower (|Ext(Expr)|.low) bound. It is computed as follows:

- |Ext(expr1) ∩ Ext(expr2)|.low = 0
  |Ext(expr1) ∩ Ext(expr2)|.high = min[|Ext(expr1)|.high, |Ext(expr2)|.high]
- |Ext(expr1) ∪ Ext(expr2)|.low = max[|Ext(expr1)|.low, |Ext(expr2)|.low]
  |Ext(expr1) ∪ |Ext(expr2)|.high = |Ext(expr1)|.high + |Ext(expr2)|.high

[11] This extensional information will be retrieved, stored and updated periodically by the system.

As a trivial case, when "expr" is the name of a term, both bounds are equal to the size of the extension of such a term.

Moreover, we use a composite measure [21] which combines the precision and recall to estimate the loss of information. We seek to measure the extension of the shaded area in Figure 2. Users may have widely varying preferences when it is necessary to choose between precision and recall. We introduce a parameter α ($0 \leq \alpha \leq 1$) to capture the preference of the user where α denotes the importance attached by a user to precision. The modified composite measure in terms of precision and recall may be given as:

$$\textbf{Loss} = 1 - \frac{1}{\alpha(\frac{1}{Precision})+(1-\alpha)(\frac{1}{Recall})}$$

Thus, a loss of information of, for example, 0.2, which is equivalent to a loss of information of 20%, means that, roughly speaking, the unwanted retrieved objects plus the wanted objects not retrieved represent the 20% of the objects presented to the user.

Notice that, as the size of the extension associated with an expression is represented by an interval, and precision, recall and loss of information metrics are based on extension sizes, then the values obtained with those metrics will also be intervals (the higher and lower bound of precision, recall and loss of information). As functions associated with those metrics increase monotonically for positive variables (the extensions of Term and Expr), it is possible to substitute both in the numerator and in the denominator the lower bound (resp. upper bound) for |Ext(Expression)| and |Ext(Term)|.

Thus, intervals (the lower and higher bound) of estimated size of extensions lead to intervals for precision and recall. We stress that intervals in precision and recall also lead to intervals for the loss of information measure. So, the real information loss of an answer presented to the user will always be between a lower and higher bound. In the following we show the two limits of the loss of information measure:

$$\text{Loss.low} = 1 - \frac{1}{\frac{1}{2}(\frac{1}{Precision.high})+\frac{1}{2}(\frac{1}{Recall.high})}$$

$$\text{Loss.high} = 1 - \frac{1}{\frac{1}{2}(\frac{1}{Precision.low})+\frac{1}{2}(\frac{1}{Recall.low})}$$

In the following we explain how the system selects the best plan among several, each one with an associated loss of information expressed as an interval. For example, let us suppose that the system has to choose between <plan1, (20%, 60%)> and <plan2, (10%, 80%)>. It is not evident which plan is the one with less loss. We never know the real loss of information *a priori* because it would require to access to the underlying data.

The system takes its decision based on the medium value corresponding to each interval. Given two plans and their associated loss of information, let us say $< plan1, (low_1, high_1) >$ and $< plan2, (low_2, high_2) >$, where low_i and $high_i$ are the lower and higher bound of the associated loss of information of $plan_i$, we define $mLoss_i = \frac{low_i + high_i}{2}$ as the medium value of the associated loss of information of $plan_i$. The following cases can arise to decide whether $plan_1$ or $plan_2$ is the plan with less loss:

(i) $mLoss_1 < mLoss_2 \Longrightarrow plan_1$ is chosen as the plan with less loss.

(ii) $mLoss_2 < mLoss_1 \Longrightarrow plan_2$ is chosen as the plan with less loss.

(iii) $mLoss_1 = mLoss_2 \Longrightarrow$ the plan with the smallest interval $(high_i - low_i)$ is chosen.

In any case, both lower bounds, low_1 and low_2, must be greater than the value defined by the user as the maximum loss allowed.

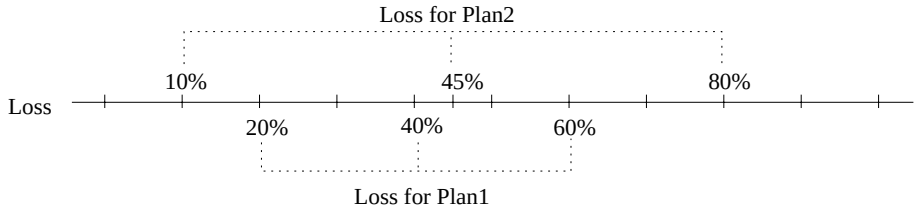

Figure 3: Intervals representing the loss of two plans

In the previous example, the medium value of plan1 (40%) is lower than the medium value of plan2 (45%), so plan1 would be chosen as the translation with less loss (see Figure 3). Anyway, notice that it would be possible than the real loss of plan2 be 10%. Other more complex probabilistic models could be used to decide among intervals but this issue is out of the scope of our work.

3.2.1. Semantic Adaptation for Precision and Recall Measures

We present here our main contribution to the estimation of the loss of information using precision and recall metrics. Although techniques on estimating precision appear in Information Retrieval literature, our work differs in the following aspects:

- We give higher priority to semantic relationships than those suggested by the underlying extensions. Only when the semantics are not available, the system resorts to the use of extensional information.
- We modify the precision/recall/information loss measures to reflect the fact that extensions are coming from different ontologies, i.e., A subsumes B does not imply that A is a superset of B if they are from different ontologies. Since the system translates a term from one ontology into an expression with terms from another different ontology with different underlying repositories, then the extensional relationships may not reflect the semantic relationships. For instance a term in a user ontology which semantically[12] subsumes a term in the target ontology may have a smaller extension than the child term.

We now enumerate the various cases that arise depending on the relationship between the conflicting term and its translation and present measures for estimating

[12]The interontology relationships used in integration of the ontologies are semantic and not extensional relationships.

the information loss. We assume that a Term is translated into an Expression in the integrated ontology. The critical step here is to estimate the extension of Expression based on the extensions of the terms in the target ontology. Precision and recall are adapted as follows:

(i) Precision and recall measures for the case where a term subsumes its translation. Semantically, the system provides a subset of the answer corresponding to the term, as Ext(Expression) $\subseteq$ Ext(Term) (by definition of subsumption). Thus, as Term subsumes Expression, we have that Ext(Term) $\cap$ Ext(Expression) = Ext(Expression). Therefore:
Precision = 1,
Recall = $\frac{|Ext(Term) \cap Ext(Expression)|}{|Ext(Term)|} = \frac{|Ext(Expression)|}{|Ext(Term)|}$

Since terms in *Expression* and *Term* are from a different ontology, the extension of *Expression* can be bigger than the extension of *Term*, although *Term* subsumes *Expression* semantically. In this case we consider the extension of *Term* to be: |Ext(Term)| = |Ext(Term) $\cup$ Ext(Expression)|. Thus, recall can be defined as:
Recall.low = $\frac{|Ext(Expression)|.low}{|Ext(Expression)|.low + |Ext(Term)|}$,
Recall.high = $\frac{|Ext(Expression)|.high}{max[|Ext(Expression)|.high, |Ext(Term)|]}$

(ii) Precision and recall measures for the case where a term is subsumed by its translation. Semantically, all elements of the term extension are returned, as Ext(Term) $\subseteq$ Ext(Expression) (by definition of subsumption). Thus, as Expression subsumes Term, we have that Ext(Term) $\cap$ Ext(Expression) = Ext(Term). The calculus of precision is similar to the one for recall in the previous case. Therefore:
Recall = 1,
Precision.low = $\frac{|Ext(Term)|}{|Ext(Expression)|.high + |Ext(Term)|}$,
Precision.high = $\frac{|Ext(Term)|}{max[|Ext(Expression)|.low, |Ext(Term)|]}$

(iii) *Term* and *Expression* are not related by any subsumption relationship. The general case is applied directly since intersection cannot be simplified. In this case the interval describing the possible loss will be wider as *Term* and *Expression* are not related semantically[13].
Precision.low = 0,
Precision.high = max[$\frac{min[|Ext(Term)|, |Ext(Expression)|.high]}{|Ext(Expression)|.high}$, $\frac{min[|Ext(Term)|, |Ext(Expression)|.low]}{|Ext(Expression)|.low}$]

Recall.low = 0, Recall.high = $\frac{min[|Ext(Term)|, |Ext(Expression)|.high]}{|Ext(Term)|}$

Given any plan it is always categorized in one of these three cases. See examples about this issue in Section 3.2.2.

We now discuss an example to justify this semantic adaptation of the metrics. Notice that if *Expr* is subsumed by *Term* and data related to *Expr* (it is the translation of Term) are retrieved, recall (how many relevant objects have been

[13] As we change in numerator and denominator we do not know which option is greater.

retrieved) cannot be zero. In fact, all the retrieved objects are relevant because of the subsumption property. And this is true even if the intersection of *Term* and *Expr* for a given extension, at a given moment, is empty. If we do not adapt those metrics, the estimated recall would be zero which is incorrect. In this context you cannot trust concrete extensions but semantic properties. Performing a semantics preserving translation does not imply that you obtain new data (that depends on the underlying data, for example, table "books" can contain no tuples), but it prevents the system from obtaining unwanted data.

In addition to the above, two special cases can arise in which the substitution of a term by an expression does not imply any loss:

(i) Substituting a term by the intersection of its immediate parents implies no loss of information if it was defined as *exactly* its definition[14], i.e., the term and the intersection of its parents are semantically equivalent.

(ii) Substituting a term by the union[15] of its children implies no loss of information if there exists an extensional relationship indicating that the term is covered extensionally by its children (total generalization).

Other semantic optimizations can be performed if overlapping and disjoint relationships are stored in the IRM repository. The union of disjoint terms is the sum of its individual sizes and the intersection is empty. And, if percentages associated with overlapping relationships are known (e.g., "20% of students are 50% of employees"), then these relationships can help to obtain a better approximation of the size corresponding to the intersection of overlapping terms.

3.2.2. Example of Translation and Measurement of the Extensional Loss

We now illustrate the computation of precision, recall and loss of information for each plan presented in Section 2.3. As the only conflicting term in the translation was 'BOOK' (the only one with no synonym into the target ontology Stanford-I), we explore the different translations for this term. For the discussions, we assume α=0.5 (equal importance to precision and recall) and the maximum loss allowed is 50%. Notice that the calculation of loss is measured as a fraction but presented to the user as a percentage value. The extensional values used in the example have been obtained from the real underlying data repositories.

(i) The loss of information incurred on substitution of 'BOOK' by 'document' is as follows; it is an example of case 2 explained in Section 3.2.1 since 'BOOK' is subsumed by 'document':

|Ext(BOOK)|=1105, |Ext(document)|=24570

$$\text{Precision.low}=\frac{|Ext(BOOK)|}{|Ext(BOOK)|+|Ext(document)|}=0.043,$$

$$\text{Precision.high}=\frac{|Ext(BOOK)|}{max[|Ext(BOOK)|,|Ext(document)|]}=0.044,$$

[14] In DL systems they are called *defined terms* and their definition specifies necessary and sufficient properties.

[15] The DL system used in the prototype lacks disjunction but other DL systems do not.

$$\text{Recall}=1,$$

$$\text{Loss.low}=1-\frac{1}{\frac{\alpha}{Precision.high}+\frac{(1-\alpha)}{Recall.high}}=0.91571,$$

$$\text{Loss.high}=1-\frac{1}{\frac{\alpha}{Precision.low}+\frac{(1-\alpha)}{Recall.low}}=0.91755$$

(ii) The loss of information incurred on substitution of 'BOOK' by 'periodical-publication' is presented in the following. It is an example of case 3 in Section 3.2.1 since 'BOOK' and 'periodical-publication' are not related semantically (none of them subsumes each other).

|Ext(BOOK)|=1105, |Ext(periodical-publication)|=34

$$\text{Precision.low} = 0,$$

$$\text{Precision.high} = \max\left[\ \frac{min[|Ext(Term)|,|Ext(Expression)|.high]}{|Ext(Expression)|.high},\ \frac{min[|Ext(Term)|,|Ext(Expression)|.low]}{|Ext(Expression)|.low}\ \right] = 1,$$

$$\text{Recall.low} = 0,$$

$$\text{Recall.high} = \frac{min[|Ext(Term)|,|Ext(Expression)|.high]}{|Ext(Term)|} = 0.03076,$$

$$\text{Loss.low}=1-\frac{1}{\frac{\alpha}{Precision.high}+\frac{(1-\alpha)}{Recall.high}}=0.94031,$$

$$\text{Loss.high}=1-\frac{1}{\frac{\alpha}{Precision.low}+\frac{(1-\alpha)}{Recall.low}}=1$$

(iii) The loss of information incurred on substitution of 'BOOK' by 'journal' is the following (another example of case 3 in Section 3.2.1):

|Ext(BOOK)|=1105, |Ext(journal)|=8

$$\text{Precision.low} = 0,$$

$$\text{Precision.high} = \max\left[\ \frac{min[|Ext(Term)|,|Ext(Expression)|.high]}{|Ext(Expression)|.high},\ \frac{min[|Ext(Term)|,|Ext(Expression)|.low]}{|Ext(Expression)|.low}\ \right] = 1,$$

$$\text{Recall.low} = 0,$$

$$\text{Recall.high} = \frac{min[|Ext(Term)|,|Ext(Expression)|.high]}{|Ext(Term)|} = 0.00723,$$

$$\text{Loss.low}=1-\frac{1}{\frac{\alpha}{Precision.high}+\frac{(1-\alpha)}{Recall.high}}=0.98564,$$

$$\text{Loss.high}=1-\frac{1}{\frac{\alpha}{Precision.low}+\frac{(1-\alpha)}{Recall.low}}=1$$

(iv) The loss of information incurred by considering the children of 'BOOK' in the integrated ontology is as follows:

|Ext(BOOK)|=1105, |Ext(book)|=14199, |Ext(proceedings)|=6, |Ext(thesis)|=0, |Ext(misc-publication)|=31, |Ext(technical-report)|=1

Ext-union.low=max[|Ext(book)|, |Ext(proceedings)|, ...]=14199,

Ext-union.high=sum[|Ext(book)|, |Ext(proceedings)|, ...]=14237

'BOOK' subsumes the union of those terms since it subsumes each of them separately, although the extension of 'BOOK' (1105) is smaller than the extension of the union (between 14199 and 14237). It is an example of case 1 in Section 3.2.1, where the extension of the subsumer is smaller than the extension of the subsumee (only possible when there are two ontologies involved with different sets of underlying data repositories).

$$\text{Ext-expr.low}=\frac{Ext-union.low}{|Ext(BOOK)|+Ext-union.low}=0.92780,$$

$$\text{Ext-expr.high}=\frac{Ext-union.high}{|Ext(BOOK)|+Ext-union.high}=0.92798,$$

$$\text{Precision}=1,$$

$$\text{Recall.low}=\frac{Ext-expr.low}{Ext-expr.low+|Ext(BOOK)|}=0.92780,$$

$$\text{Recall.high}=\frac{Ext-expr.high}{max[|Ext(BOOK)|,Expr-ext.high]}=^{16}\ 1,$$

$$\text{Loss.low}=1-\frac{1}{\frac{\alpha}{Precision.high}+\frac{(1-\alpha)}{Recall.high}}=0,$$

$$\text{Loss.high}=1-\frac{1}{\frac{\alpha}{Precision.low}+\frac{(1-\alpha)}{Recall.low}}=0.07220$$

Thus, the four possible plans and the respective losses for the user query '*(***AND** *BOOK (***FILLS** *doc-author-name "Carl Sagan"))*' are illustrated in Table 2.

Plan	Loss of Information
(**AND** document (**FILLS** doc-author-name "Carl Sagan"))	91.57%<loss<91.75%
(**AND** periodical-publication (**FILLS** doc-author-name "Carl Sagan"))	94.03%<loss<100%
(**AND** journal (**FILLS** doc-author-name "Carl Sagan"))	98.56%<loss<100%
(**AND** UNION(book, proceedings, thesis, misc-publication, technical-report) (**FILLS** doc-author-name "Carl Sagan"))	0%<loss<7.22%

Table 2: The various plans and the associated loss of information

Only the fourth case fulfils the condition about keeping the loss of information below the maximum loss allowed (50%) and is hence chosen. That means that the chosen translation of the original user query '*[NAME PAGES] for (***AND** *BOOK (***FILLS** *CREATOR "Carl Sagan"))*' is '*[title number-of-pages] for (***AND** *UNION(book, proceedings, thesis, misc-publication, technical-report) (***FILLS** *doc-author-name "Carl Sagan"))*'. The answer does not contain incorrect data in the best case (which is possible) but, in the worst case, around 7% of the ideal answer may be missed (substituted by irrelevant data or not accessed).

[16] If the higher bound is 1 or the lower bound is 0, then no new information has been obtained.

4. Local Decision vs. Global Decision for Choosing the Optimal Plan

As we mentioned in Section 2, we propose a method which looks for all possible translations (plans) and then chooses the one with the least loss of information. One could think that a way to improve the performance is to decide at each step (for each non-translated term) whether it is better to translate using the intersection of its parents or using the union of its children. This is a case of making a local decision, as opposed to a global one after generating all possible translations. We show in the following that taking local decisions may result in the choice of a non-optimal translation.

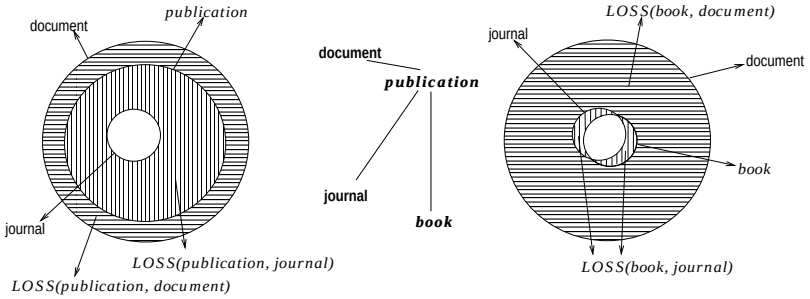

Figure 4: Counterexample for local decision vs. global decision

Consider Figure 4, where 'book' and 'publication' are terms from the user ontology and 'document' and 'journal' are terms from the target ontology. Notice how inner circles corresponds to subsumed terms. As 'journal' does not subsume 'book' neither 'book' subsumes 'journal' then their respective circles can overlap partially or even not at all. Let us consider the case in which 'book' is a conflicting term that has to be translated. Here it is substituted by its parents, 'publication', which should be substituted by 'document' or by 'journal'. Let *LOSS(X, Y)* be the loss of information incurred when X is substituted by Y. As identified in the figure, on the left, LOSS(publication, document) (Horizontal Shading) < LOSS(publication, journal) (Vertical Shading). Thus 'document' would be chosen as the best translation of 'publication' and therefore 'document' would also be taken as the translation of 'book', the original conflicting term. But we can observe on the right side of the figure that LOSS(book, document) (Horizontal Shading) > LOSS(book, journal) (Vertical Shading). This means that 'journal' is the best translation for 'book' although the best translation for 'publication' is 'document'.

This case arises because 'publication' and 'document' are very close extensionally and semantically ('publication' and 'document' circles are very similar) and the same is true for 'book' and 'journal' ('book' and 'journal' circles are very similar too). At the same time both "pairs" are quite far from each other extensionally and semantically (see in the figure how the circles named 'publication' and 'document' are much bigger than the circles 'book' and 'journal', i.e., 'publication' and 'document' are much more general than 'book' and 'journal'). In the hierarchy we have tried to represent this idea by placing similar abstractions at a similar height.

Every time this happens, taking local decisions is a mistake and the system would not choose the translation with less loss correctly.

Furthermore, a recursive method that takes local decisions would need to calculate the loss of information at each step by combining precision and recall of previous stages. This technique was rejected since the extensional information of conflicting terms, which are not the original conflicting term (they are, for instance, parents of the original conflicting term), should not be taken into account. For instance, in the example shown in Figure 4, where 'book' is the conflicting term and 'publication' is a parent of 'book' (belonging both to the same user ontology), a local decision at 'publication' would choose between translating 'publication' by 'document' or by 'journal'. The estimation of the loss incurred would imply the use of the size of the extensions of 'document', 'journal' and 'publication', as we have seen in previous sections. But the extension of 'publication' will never be accessed since the problem considered is to translate 'book'. As they are in the same ontology, by providing the objects of 'publication' we are not adding any new object if we already accessed 'book'. On the contrary, as 'document' and 'journal' are terms of the target ontology (with different underlying repositories than the ones under the user ontology), providing the objects under 'document' or under 'journal' can enrich the current answer about 'book'. Of course, each case has an associated loss, so what the system has to do is to choose between 'document' and 'journal' to obtain the translation with less loss with respect to 'book' (the same decision with respect to 'publication' is not relevant).

5. Loss of Information for Correlated Answer across Ontologies

After the plan with the least loss for the conflicting terms is chosen, the corresponding data will be retrieved from the data repositories underlying the target ontology. In the translation process, the system takes care of keeping the loss of information corresponding to the new data under the maximum loss defined by the user. After accessing the data corresponding to the best plan, the system calculates the *real* loss of information associated with the new answer which will be inside the interval obtained in the estimation of the loss. Notice that now the system does not need to approximate the extension of the translation since the data has been already accessed and the unions and intersections have been performed. This explains why we call it the *real* loss of information.

Suppose that we deal with $Answer_1$, obtained with the user query ($Query_1$) executed over the user ontology (with no loss); and with $Answer_2$ coming from the first target ontology (with a certain loss of information) that corresponds to $Query_2$ (the best translation of $Query_1$ into the first target ontology). Both answers are correlated by performing a union operation in order to present to the user a combined answer. The loss of information associated with $Answer_2$ was calculated based on the estimation corresponding to answering $Query_2$ instead of $Query_1$. Therefore, the associated loss of information of the correlated answer can be calculated by using the same mechanisms explained in Section 3.2 because the substitution performed has been to use $Query_1 \cup Query_2$ (let us call it, $NewQuery_2$). Thus

the system obtains a new answer $NewAnswer_2 = Answer_1 \cup Answer_2$ as we have already said[17]. Remember that the system knows the exact size of extensions of $Answer_1$ and $Answer_2$.

From this step, and until the user is satisfied with the answer, two alternatives can arise when enriching the answer:

- The new plan ($Query_3$) used to enrich the answer is one of the previously calculated ones when visiting some target ontology (it will be the one with less loss among those plans that keep the loss below the maximum loss allowed). In this case, the same target ontology is accessed again, this time to retrieve the data corresponding to $Query_3$. Notice that the new answer $Answer_3$ and $Answer_2$ have been obtained from the same ontology. Some special cases can arise:

  (i) If one of the two answers has an associated recall=1 the system cannot improve the quality of the corresponding answer. A union would decrease the precision and an intersection would decrease the recall. In this case the new plan is rejected and a new plan is chosen by the system. This check can actually be performed before accessing the data under $Query_3$.

  (ii) If both answers have recall=1 then an intersection does not decrease the recall but increases the precision (less unwanted data).

  $$NewAnswer_3 = Answer_1 \cup (Answer_2 \cap Answer_3)$$
  ($Answer_2$ and $Answer_3$ are plans from the same ontology)

  (iii) If both answers have a recall < 1 (the most common case), then the union between them is performed in order to increase the recall in spite of (probably) decreasing the precision.

  $$NewAnswer_3 = NewAnswer_2 \cup Answer_3$$
  (Notice that $Answer_2$ is included in $NewAnswer_2$)

- No stored plan can be used, so a new target ontology is chosen and a new set of plans is obtained. Let $Query_3$ be the one with less loss. The corresponding underlying data, $Answer_3$, will be correlated with the previous answer by performing a union operation.

  $$NewAnswer_3 = NewAnswer_2 \cup Answer_3$$

Each time the system correlates two answers (the one presented previously to the user and the new one), both with a certain loss of information associated, although both answers keep the loss under the maximum loss allowed, it can happen that *the correlated answer has a loss of information greater than the maximum loss defined by the user*. This can happen because in the correlated answer (obtained through a union operation) the precision may be reduced much more than the increase in

[17] $Query_i$ denotes the plan used in iteration i, and $Answer_i$ is the data corresponding to that $Query_i$. $NewAnswer_i$ is the correlated answer in the iteration i that corresponds to the plan $NewQuery_i$ (the union of all the plans used until iteration i).

recall. In other words, although the correlated answer contains more relevant data, it also contains much more unwanted data compared to before correlation. In Figure 5 we can observe how $NewAnswer_{i+1} = NewAnswer_i \cup Answer_{i+1}$ (on the right) has much more associated loss (shaded area) than $NewAnswer_i$ and $Answer_{i+1}$ individually (on the left and middle).

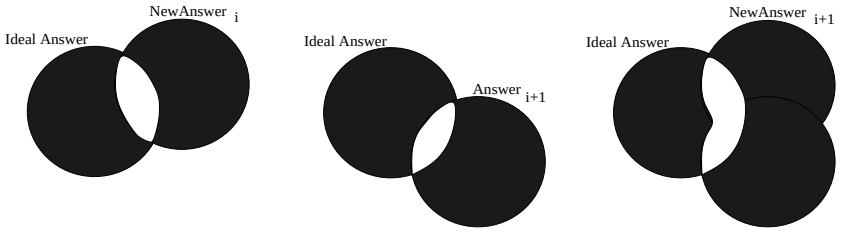

Figure 5: Loss of information of a correlated answer

Thus, after obtaining the correlated answer the loss for this answer is then calculated (instead of $Query_1$ the system has calculated the data corresponding to $NewQuery_{i+1}$, i.e., the union of all the plans used until iteration $i+1$). If this loss exceeds the maximum loss allowed then the correlated answer is rejected and only the previous and the new answers are shown to the user separately (with the corresponding warning). Future new answers could be correlated to one of the two answers (always keeping the loss of the correlated answer below the limit). If at any time the user allows a greater loss, then the different answers could be correlated into one answer satisfying the restriction set by the user.

6. Conclusions

As the Web becomes the predominant environment for more and more people to create applications, and export or share information, syntactic approaches for navigation and keyword based searches are becoming increasingly inadequate. We present a novel approach based on the use of multiple, possibly pre-existing, real world domain ontologies as views on the underlying data repositories. Thus, an information request can now be expressed using terms from these ontologies and a system can now browse multiple domain ontologies as opposed to users browsing individual heterogeneous repositories or web pages correlated based on statistical information.

The main contribution of this paper is the characterization of the *loss of information* when a translation results in a change of semantics of the query. Measures to estimate loss of information based on terminological difference as well as on standard and well accepted measures, such as *precision* and *recall*, are also presented. As far as we know, our work is the first that deals with the problem of measuring the imprecision of answers in the context of managing multiple distributed and heterogeneous data repositories.

Approaches for modeling imprecision and measures for uncertain information have been proposed in the literature. The novelty of our approach is that we provide a set theoretic basis for an extensional measure of semantic information loss. The

user is provided with a means to control the quality of information based on his preference of more precision or more recall, and the limit of the total loss incurred. Furthermore, a qualitative description of information loss using intensional term descriptions is also presented and illustrated with the help of examples. Based on the estimates of information loss, the system chooses that translation which minimizes the loss of information. We thus establish vocabulary heterogeneity as the basis for identifying and measuring the quality of information, a very useful feature for information processing in open and dynamic environments.

Experimenting with the implemented system, we found cases where, after visiting several ontologies, the user did not obtain a single wanted data even when data satisfying the user requirements were stored in some of the underlying data repositories described by the visited ontologies. The main reasons for that were: first, some ontologies only modeled their underlying data repositories partially; and second, different points of view were used to describe the same conceptualizations. However, by allowing a controlled relaxation of the precision for the same cases, data were obtained and, in fact, they did satisfy all the constraints in the user query, although some constraints were not explicitly verified by the system. Thus, the real loss of information associated with those "imprecise" answers was 0%. Therefore, we conclude by stressing the importance of dealing with imprecise answers in query processing for Global Information Systems, where there is likely to be significant variations in modeling and semantics.

References

[1] Y. Arens, C.A. Knoblock, and W. Shen. Query Reformulation for Dynamic Information Integration. *Journal of Intelligent Information Systems*, 6(2-3):99–130, 1996.

[2] R. Bayardo et al. InfoSleuth: Agent-Based Semantic Integration of Information in Open and Dynamic Environments. In *Proceedings of the 1997 ACM International Conference on the Management of Data (SIGMOD), Tucson, Arizona.*, May 1997.

[3] J.M. Blanco, A. Goñi, and A. Illarramendi. Mapping among Knowledge Bases and Data Repositories: Precise Definition of its Syntax and Semantics. *Information Systems*, 24(4):275–301, 1999.

[4] T. Catarci and M. Lenzerini. Representing and using interschema knowledge in cooperative information systems. *International Journal on Cooperative Information Systems*, 2(4), 1993.

[5] S. Chaudhuri. Generalization and a Framework for Query Modification. In *Proceedings of the sixth International Conference on Data Engineering*, February 1990.

[6] S. Chawathe, H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. Ullman, and J. Widom. The TSIMMIS Project: Integration of Heterogeneous Information Sources. In *Proceedings of 10th IPSJ conference, Tokyo, Japan*, October 1994.

[7] W. W. Chu, H. Yang, K. Chiang, M. Minock, G. Chow, and C. Larson. Cobase: A Scalable and Extensible Cooperative Information System. *Journal of Intelligent Information Systems*, 6(2-3), 1996.

[8] C. Collet, M. N. Huhns, and W. Shen. Resource Integration Using a Large Knowledge Base in CARNOT. *IEEE Computer*, 24(12):55–62, December 1991.

[9] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Andrea Schaerf. Reasoning in description logics. In Gerhard Brewka, editor, *Principles of Knowledge Representation*, Studies in Logic, Language and Information, pages 193–238. CSLI Publications, 1996.

[10] D. Dubois, J. Lang, and H. Prade. Automated Reasoning using Possibilistic Logic: Semantics, Belief Revision, and Variable Certainty Weights. *IEEE Transactions on Knowledge and Data Engineering*, 6(1), February 1994.

[11] A. Goñi, E. Mena, and A. Illarramendi. Querying Heterogeneous and Distributed Data Repositories using Ontologies. In *Proceedings of the 7th European-Japanese Conference on Information Modelling and Knowledge Bases (IMKB'97), Toulouse (France)*. IOS Press, ISBN 905199396X, May 1997.

[12] T. Gruber. A translation Approach to Portable Ontology Specifications. *Knowledge Acquisition, An International Journal of Knowledge Acquisition for Knowledge-Based Systems*, 5(2), June 1993.

[13] J. Hammer and D. McLeod. An approach to resolving Semantic Heterogeneity in a Federation of Autonomous, Heterogeneous, Database Systems. *International Journal on Intelligent and Cooperative Information Systems*, 2(1), March 1993.

[14] V. Kashyap and A. Sheth. Semantic and Schematic Simililarities between Databases Objects: A Context-based approach. *The VLDB Journal*, 5(4), December 1996.

[15] A.Y. Levy, D. Srivastava, and T. Kirk. Data Model and Query Evaluation in Global Information Systems. *Journal of Intelligent Information Systems*, 5(2):121–143, September 1995.

[16] E. Mena. *OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies*. PhD thesis, University of Zaragoza, November 1998. http://siul02.si.ehu.es/PUBLICATIONS/thesis98.ps.gz.

[17] E. Mena, A. Illarramendi, V. Kashyap, and A. Sheth. OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies. *To be published in the journal Distributed And Parallel Databases (DAPD)*, 1999.

[18] E. Mena, V. Kashyap, A. Illarramendi, and A. Sheth. Domain Specific Ontologies for Semantic Information Brokering on the Global Information Infrastructure. In *Proc. of the International Conference on Formal Ontologies in Information Systems (FOIS'98). Trento (Italy)*. IOS Press, ISBN 0922-6389, June 1998.

[19] E. Mena, V. Kashyap, A. Sheth, and A. Illarramendi. OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies. In *Proc. of the First IFCIS International Conference on Cooperative Information Systems (CoopIS'96), Brussels (Belgium), June*. IEEE Computer Society Press, 1996.

[20] A. Motro. Multiplex: A Formal Model of Multidatabases and its Implementations. Technical report, Technical Report ISSE-TR-95-103, Department of Information and Software Systems Engineering, George Mason University, Fairfax, Virginia, March 1995.

[21] C.J. Rijsbergen. *Information Retrieval*. London: Butterworths, 1979. http://www.dcs.gla.ac.uk/Keith/Chapter.7/Ch.7.html.

[22] G. Salton. *Automatic Text Processing*. Addison-Wesley, 1989.

[23] A. Sheth and J. A. Larson. Federated Database Systems for Managing Distributed Heterogeneous and Autonomous Databases. *ACM Computing Surveys*, 22(3):183–236, September 1990.

[24] A. Silberschatz and S. Zdonik. Database Systems - Breaking Out of the Box. *SIGMOD Record*, 26(3), September 1997.

[25] Pauray S. M. Tsai and Arbee L. P. Chen. Querying Uncertain Data in Heterogeneous Databases. In *Third International Workshop on Research Issues in Data Engineering: Interoperability in Multidatabase Systems, Vienna, Austria*, April 1993.