UMR IRISA

Project-Team PILGRIM

Gradedness, Imprecision, and Mediation in Database Management Systems

Lannion

Activity Report

2009

1 Team

Head of the team
Olivier Pivert, Professor, Enssat

Administrative assistant
Joëlle Thépault, Enssat, 20%
Nelly Vaucelle, Enssat, 10%

Université Rennes 1 personnel
Patrick Bosc, Professor, Enssat
Allel Hadjali, Associate Professor, Enssat
Hélène Jaudoin, Associate Professor, Enssat
Ludovic Liétard, Associate Professor, IUT Lannion
Daniel Rocacher, Associate Professor, HdR, Enssat
Grégory Smits, Associate Professor, IUT Lannion

PhD students
Amine Mokhtari, Région Bretagne grant and Conseil Général 22 grant, since October 2007
Nouredine Tamani, Région Bretagne grant and Conseil Général 22 grant, since January 2008

2 Overall Objectives

2.1 Introduction

In a majority of works undertaken in the area of database management systems (DBMSs), it is assumed that data are perfectly known and that only Boolean queries can be expressed. The aim of the project is to soften this double hypothesis by introducing the notions of imprecision and graduality in information systems. Such concepts impact at least two aspects of database systems:

1. the expression of queries addressed to DBMSs, which become gradual or flexible. The idea is to allow the user to specify preferences instead of Boolean conditions, thus leading to a result whose elements are ordered accordingly, e.g., find the *young* employees who work in a *high* budget department,

2. the description and the manipulation of imprecisely known data. Such data can be expressed in a linguistic way, e.g., a reservoir which is *big* or a person who is *young*.

The project reconsiders several aspects of DBMSs by taking imprecision and graduality into account. From a querying point of view, these two characteristics are indeed orthogonal, i.e., one may consider regular queries addressed to databases containing imprecise data, or flexible queries against regular databases. The originality of the project is to rely on a common scientific framework, namely fuzzy sets, to deal with these two sides and then to be able to consider the "joint" issue of flexible queries addressed to databases containing imprecise data.

The issue of flexible querying has received an increasing attention in the last years in the database community and the idea of including preferences inside queries gets more and more acceptance. The line followed in Pilgrim is to focus on:

1. various types of flexible conditions, including non-trivial ones,
2. the semantics of such conditions from a user standpoint,
3. the design of query languages providing flexible capabilities in a relational setting.

Basically, a flexible query involves linguistic terms corresponding to gradual predicates, i.e., predicates which are more or less satisfied by a given (attribute) value. In addition, these various terms may have different degrees of importance, which means that they may be connected by operators beyond conjunction and disjunction. For instance, in the context of a search for used vehicles, a user might say that he/she wants a *compact* car *preferably French*, with a *medium* mileage, *around* 6 k$, whose color is *as close as possible* to light grey or blue. The terms appearing in this example must be specified, which requires a certain theoretical framework. For instance, one may think that "*preferably* French" corresponds to a complete satisfaction for French cars, a lower one for Italian and Spanish ones, a still smaller satisfaction for German cars and a total rejection for others. Similarly, "*medium* mileage" can be used to state that cars with less than 40000 km are totally acceptable while the satisfaction decreases as the mileage goes up to 75000 km which is an upper bound. Moreover, it is likely that some of the conditions are more important than others (e.g., the price with respect to the color). In such a context, answers are ordered according to their overall compliance with the query, which makes a major difference with respect to usual queries.

In the previous example, conditions are fairly simple, but it turns out that more complex ones can also intervene. A particular attention is paid to conditions calling on aggregate functions together with gradual predicates. For instance, one may look for departments where *most* employees are *close* to retirement, or where the average salary of *young* employees is *around* $2500. Such statements have their counterpart in regular query language, such as SQL, and the specification of their semantics, when gradual conditions come into play, is studied in the project.

Along this line, the ultimate goal of the project is to introduce gradual predicates inside database query languages, thus providing flexible querying capabilities. Algebraic languages as well as more user-oriented languages are under consideration in both the original and extended relational settings.

Since a few years, another important line of research concerns the querying of databases in the presence of imprecise data. We think that this type of database should receive more and more attention in the future, for instance in the context of automated recognition, data fusion, forecasts or incomplete archives, in which imprecision is intrinsically present. Some works in the 80s have established that null values and disjunctive data have a strong impact on the queries that can be processed in an efficient way. This situation remains true in the context considered in Pilgrim, where the objective is to investigate various types of queries that can be processed efficiently in the relational setting. Although there is no hope for a general family of queries which would include all of the operators of the relational algebra, the identification of its largest subset which fits our requirements is undertaken. Beyond this target, specific queries that make sense and are tractable are sought, such as queries of the type: "to what extent is it possible that tuple t belongs to the result of query Q", where t is a given tuple and Q is an algebraic query. Such queries are called possibilistic queries (due to the presence of the word "possible" in their statement) and they stem from regular yes/no queries.

Querying imprecise data calls on a given theory of uncertainty. Possibility theory is mainly used in Pilgrim, essentially because it provides a framework which is coherent with the one serving for flexible queries. By doing so, it becomes possible to study the issue of flexible queries against imprecise databases. It is worth noticing that the case where probabilistic data are used instead of possibilistic ones, is also a matter of interest.

A new research topic in Pilgrim concerns flexible data integration systems. One considers a distributed database environment where several data sources are available. An extreme case is that of a totally decentralized P2P system. An intermediary situation corresponds to the case where a global schema is available and where the sources can be accessed through views defined on that schema (LAV approach). The problem consists in handling a user query (possibly involving preferences conveyed by fuzzy terms) so as to forward it (or part of it) to the relevant data sources, after rewriting it into the "language" of each selected source. The overall objective is thus to define flexible semantic mapping mechanisms and flexible query rewriting techniques which take into account both the approximate nature of the mappings and the graded nature of the initial query. A large scale environment is aimed, and the performance aspect is therefore crucial in such a context.

2.2 Highlights of the Year

- Organization of the first French-speaking Conference on Information Technologies, Communication and Geolocation in Transportation Systems (CoGIST'09), held in Saint-Quay-Portrieux, June 29 – July 1, 2009.
- Organization of a special session devoted to Advances in Soft Computing Applied to Databases and Information Systems at the international conference IFSA-EUSFLAT 2009.
- Joint work with IRIT (Toulouse) which led to the proposal of a new model for uncertain databases, based on the notion of a possibilistic certainty level.

3 Scientific Foundations

The project investigates the issues of flexible queries against regular databases as well as regular queries addressed to databases involving imprecise data. These two aspects make use of two close theoretic settings: fuzzy sets for the support of flexibility and possibility theory for the representation and treatment of imprecise information.

3.1 Fuzzy sets

Fuzzy sets were introduced by L.A. Zadeh in 1965 [Zad65] in order to model sets or classes whose boundaries are not sharp. This is particularly the case for many adjectives of the natural language which can be hardly defined in terms of usual sets (e.g., high, young, small, etc.), but are a matter of degree. A fuzzy (sub)set F of a universe X is defined thanks to a membership function denoted by μ_F which maps every element x of X into a degree $\mu_F(x)$ in the unit interval $[0, 1]$. When the degree equals 0, x does not belong at all to F, if it is 1, x is a full member of F and the closer $\mu_F(x)$ to 1 (resp. 0), the more (resp. less) x belongs to F. Clearly, a regular set is a special case of a fuzzy set where the values taken by the membership function are restricted to the pair $\{0,\ 1\}$. Beyond the intrinsic values of the degrees, the membership function offers a convenient way for ordering the elements of X and it defines a symbolic-numeric interface. The α level-cut of a fuzzy set F is defined as the (regular) set of elements whose degree of membership is greater than or equal to α and this concept bridges fuzzy sets and ordinary sets.

Similarly to a set A which is often seen as a predicate (namely, the one appearing in the intentional definition of A), a fuzzy set F is associated with a gradual (or fuzzy) predicate. For instance, if the membership function of the fuzzy set *young* is given by: $\mu_{young}(x) = 0$ for any $x \geq 30$, $\mu_{young}(x) = 1$ for any $x < 21$, $\mu_{young}(21) = 0.9$, $\mu_{young}(22) = 0.8$, ... , $\mu_{young}(29) = 0.1$, it is possible to use the predicate *young* to assess the extent to which Tom, who is 26 years old, is young ($\mu_{young}(26) = 0.4$).

The operations valid on sets (and their logical counterparts) have been extended to fuzzy sets. Their definition assumes the validity of the commensurability principle between the concerned fuzzy sets. It has been shown that it is impossible to maintain all of the properties of the Boolean algebra when fuzzy sets come into play. Fuzzy set theory starts with a strongly coupled definition of union and intersection which rely on triangular norms ($\top$) and co-norms ($\bot$) tied by de Morgan's laws. Then:

$$\mu_{A \cap B}(x) = \top(\mu_A(x), \mu_B(x)) \qquad \mu_{A \cup B}(x) = \bot(\mu_A(x), \mu_B(x))$$

The complement of a fuzzy set F, denoted by $\bar{F}$, is a fuzzy set such that: $\mu_{\bar{F}}(x) = neg(\mu_F(x))$, where *neg* is a strong negation operator and the complement to 1 is generally used. The conjunction and disjunction operators are the logical counterpart of intersection and union while the negation is the counterpart of the complement.

In practice, minimum and maximum are the most commonly used norm and co-norm because they have numerous properties among which:

- the satisfaction of all the properties of the usual intersection and union (including idempotency and double distributivity), except excluded-middle and non-contradiction laws,
- they still work with an ordinal scale, which is less demanding than numerical values over the unit interval,
- the simplicity of the underlying calculus.

[Zad65] L. Zadeh, "Fuzzy sets", *Information and Control 8*, 1965, p. 338–353.

Once these three operators given, others can be extended to fuzzy sets, such as the difference:

$$\mu_{E-F}(x) = \top(\mu_E(x), \mu_{\bar{F}}(x))$$

and the Cartesian product:

$$\mu_{E\times F}(x, y) = \top(\mu_E(x), \mu_F(y)).$$

The inclusion can be applied to fuzzy sets in a straightforward way: $E \subseteq F \Leftrightarrow \forall x, \mu_E(x) \leq \mu_F(x)$, but a gradual view of the inclusion can also be introduced. The idea is to consider that E may be more or less included in F. Different approaches can be envisaged, among which one is based on the notion of a fuzzy implication (the usual logical counterpart of the inclusion). The starting point is the following definition valid for sets:

$$E \subseteq F \Leftrightarrow \forall x, x \in E \Rightarrow x \in F$$

which becomes :

$$deg(E \subseteq F) = \top_x(\mu_E(x) \Rightarrow_f \mu_F(x))$$

where $\Rightarrow_f$ is a fuzzy implication whose arguments and result take their value in the unit interval. Different families of such implications have been identified (notably R-implications and S-implications) and the most common ones are:

- Kleene-Dienes implication : $a \Rightarrow_{K-D} b = max(1 - a, b)$,
- Rescher-Gaines implication: $a \Rightarrow_{R-G} b = 1$ if $a \leq b$ and 0 otherwise,
- Gödel implication : $a \Rightarrow_{Go} b = 1$ is $a \leq b$ and b otherwise,
- Lukasiewicz implication : $a \Rightarrow_{Lu} b = min(1, 1 - a + b)$.

Of course, fuzzy sets can also be combined in many other ways, for instance using mean operators, which do not make sense for classical sets.

3.2 Possibility theory

Possibility theory is a theory of uncertainty which aims at assessing the realization of events. The main difference with the probabilistic framework lies in the fact that it is mainly ordinal and it is not related with frequency of experiments. As in the probabilistic case, a measure (of possibility) is associated with an event. It obeys the following axioms [Zad78]:

[Zad78] L. Zadeh, "Fuzzy sets as a basis for a theory of possibility", *Fuzzy Sets and Systems 1*, 1978, p. 3–28.

- $\Pi(X) = 1$,
- $\Pi(\oslash) = 0$,
- $\Pi(A \cup B) = max(\Pi(A), \Pi(B))$,

where X denotes the set of all events and A, B are two subsets of X. If $\Pi(A)$ equals 1, A is completely possible (but not certain), when it is 0, A is completely impossible and the closer to 1 $\Pi(A)$, the more possible A. From the last axiom, it appears that the possibility of $\bar{A}$, the opposite event of A, cannot be calculated from the possibility of A. The relationship between these two values is:

$$max(\Pi(A), \Pi(\bar{A})) = 1$$

which stems from the first and third axioms (where B is replaced by $\bar{A}$).

In other words, if A is completely possible, nothing can be deduced for $\Pi(\bar{A})$. This state of fact has led to introduce a complementary measure (N), called necessity, to assess the certainty of A. $N(A)$ is based on the fact that A is all the more certain as $\bar{A}$ is impossible [DP80]:

$$N(A) = 1 - \Pi(\bar{A})$$

and the closer to 1 N(A), the more certain A. From the third axiom on possibility, one derives:

$$N(A \cap B) = min(N(A), N(B)))$$

and, in general:

- $\Pi(A \cap B) \leq min(\Pi(A), \Pi(B))$,
- $N(A \cup B) \geq max(N(A), N(B))$.

In the possibilistic setting, a complete characterization of an event requires the computation of two measures: its possibility and its certainty. It is interesting to notice that the following property holds:

$$\Pi(A) < 1 \Rightarrow N(A) = 0.$$

It indicates that if an event is not completely possible, it is excluded that it is somewhat certain, which makes it possible to define a total order over events: first, the events which are somewhat possible but not at all certain (from (Π = N = 0 to Π = 1 and N = 0), then those which are completely possible and somewhat certain (from Π = 1 and N = 0 to Π = N = 1).

[DP80] D. Dubois, H. Prade, *Fuzzy set and systems: theory and applications*, Academic Press, 1980.

This favorable situation (existence of a total order) is valid for usual events, but if fuzzy ones are taken into account, this is no longer true (because A $\cup$ $\bar{A} = X$ is not true in general when A is a fuzzy set) and the only valid property is: $\forall$ A, $\Pi(A) \geq \mathrm{N}(A)$.

The notion of a possibility distribution [Zad78], denoted by π, plays a role similar to that of a probability distribution. It is a function from the referential X into the unit interval and:

$$\forall A \subseteq X, \Pi(A) = sup_{x \in A} \pi(x)$$

In order to comply with the second axiom above, a possibility distribution must be such that there exists (at least) an element x_0 of X for which $\pi(x_0) = 1$. Indeed, a possibility distribution can be seen as a normalized fuzzy set F which represents the knowledge about a given variable. The following formula:

$$\pi(x = a) = \mu_F(a)$$

which is often used, tells that the possibility that the actual value of the considered variable x is a, equals the degree of membership of a to the fuzzy set F. For example, Paul's age may be only imprecisely known as "close to 20", where a given fuzzy set is associated with this fuzzy linguistic expression.

3.3 Fuzzy sets, possibility theory and databases

The project is situated at the crossroads of databases and fuzzy sets. Its main objective is to broaden the capabilities offered by DBMSs according to two orthogonal lines in order to separate two distinct problems:

- flexible queries against regular databases so as to provide users with a qualitative result made of ordered elements,
- Boolean queries addressed to databases containing imprecise attribute values.

Once these two aspects solved separately, the joint issue of flexible queries against databases containing imprecise attribute values will also be considered. This can be envisaged because of the compatibility between the semantics of grades (preferences) in both fuzzy sets and possibility distributions.

It turns out that fuzzy sets offer a very convenient way for modeling gradual concepts and then flexible queries [1]. It has been proven [BP92] that many *ad hoc* approaches (e.g., based on distances) were special cases of what is expressible using fuzzy set theory. This framework makes it possible to express sophisticated queries where the semantic choices of the user can take place (e.g., the meaning of the terms or the compensatory interaction desired between the various fuzzy conditions of a query). The works conducted in Pilgrim aim at extending

[Zad78] L. Zadeh, "Fuzzy sets as a basis for a theory of possibility", *Fuzzy Sets and Systems 1*, 1978, p. 3–28.

[BP92] P. Bosc, O. Pivert, "Some approaches for relational databases flexible querying", *Journal of Intelligent Information Systems 1*, 1992, p. 323–354.

algebraic as well as user-oriented query languages in both the relational and the object-oriented (extended relational in practice) settings. The relational algebra has already been revised in order to introduce flexible queries and a particular focus has been put on the division operation. Current works are oriented towards:

- conditions calling on aggregate functions applying to fuzzy sets, for instance fuzzy quantified statements such as "most employees have a medium salary" which can be expressed in the context of an SQL-like language,
- the handling of fuzzy bags (fuzzy multisets) and their connection with fuzzy numbers.

Recent advanced developments on flexible querying in Databases Management Systems and Information Retrieval Systems in the literature can be found in reference [1] where innovative ideas in this area are discussed.

As to possibility distributions, they are used to represent imprecise (imperfect) data. So doing, a straightforward connection can be established between a possibilistic database and regular ones. Indeed, a possibilistic database is nothing but a weighted set of regular databases (called worlds), obtained by choosing one candidate in every distribution appearing in any tuple of every possibilistic relation. According to this view, a query addressed to a possibilistic database has a natural semantics. However, it is not realistic to process it against all the worlds due to their huge number. Then, the question tied to the querying of a possibilistic database bears mainly on the efficiency, which imposes to obviate the combinatory explosion of the worlds. The objective of the project is to identify different families of queries which comply with this requirement in the context of the relational setting, even if the initial model must obviously be extended (in particular to support imprecise data).

3.4 Query rewriting using views

Information integration is the problem of combining information residing at disparate sources and providing the user with a unified view of that information. This problem has been a long standing challenge for the database community.

Two main approaches for information integration have been proposed. In the first approach, namely materialization or warehousing, data are periodically extracted from the sources and stored in a centralized repository, called a (data) warehouse. User queries are posed and executed at the warehouse with no need to access the remote information sources. Such an approach is useful in the context of intra-enterprise integration with few remote sources to integrate. It is, however, not feasible in open environments like the Web where the number of sources may be very large and dynamic.

In the second approach, called mediation or virtual integration, data stay at the sources and are collected dynamically in response to user queries [Len02,Hal03]. Mediation architectures are based on the mediator/wrapper paradigm where native information sources are *wrapped* into

[Len02] M. Lenzerini, "Data Integration : A Theoretical Perspective", Madison, Wisconsin, 2002.

[Hal03] A. Halevy, "Data Integration : A status Report", *in : German Database Conference BTW-03*, Leipzig, Germany, 2003. Invited Talk.

logical views through which the underlying sources may be accessed. The views are stored in the mediator component which additionally contains an integrated global schema that provides a single entry point to query the available information sources. The global schema acts as an interface between the user queries and the sources, freeing the users from the problem of source location and heterogeneity issues. In such an architecture, user queries posed on the global schema are rewritten in terms of logical views and then sent to the remote sources.

Briefly stated, two main approaches of mediation have been investigated [Hal01]: the GAV (Global As View) approach where the global schema is expressed as a set of views over the data sources, and the LAV (Local As View) approach where the data sources are defined as views over the global schema. Query processing is expected to be easier in the GAV approach as it can be achieved by a kind of unfolding of original queries. However, this approach suffers from a lack of extensibility as changing or adding new sources affects the global schema. On the contrary, the LAV approach is known to be highly extensible in the sense that source changes do not impact the global schema. However, in the context of the LAV approach, query processing is known to be more challenging.

A centralized mediation approach has several drawbacks including scalability, flexibility, and availability of information sources. To cope with such limitations, a new decentralized integration approach, based on a Peer-to-Peer (P2P) architecture, has been proposed. A P2P data management system [HIM+04] enables sharing heterogeneous data in a distributed and scalable way. Such a system is made of a set of peers each of which is an entire data source with its own distinct schema. Peers interested in sharing data can define pairwise mappings between their schemas. Users formulate queries over a given peer schema then a query answering system exploits relevant mappings to reformulate the original query into set of queries that enable to retrieve data from other peers.

Query answering in information integration systems

The problem of answering queries in mediation systems has been intensively investigated during the last decade. In particular, the investigation of this problem in the context of a LAV approach led to a great piece of fundamental theory. Recent works show that query processing in data integration is related to the general problem of answering queries using views [Hal01, Len02]. In such a setting, the semantics of queries can be formalized in terms of certain answers [AD98]. Intuitively, a certain answer to a query Q over a global (mediated) schema with respect to a set of source instances is an answer to Q in any database over the global schema that is consistent with the source instances. Therefore, the problem of answering queries in LAV-based mediation systems can be formalized as the problem of computing all the certain answers to the queries. As shown recently, this problem has a strong connection with the problem of query answering in database with incomplete information under constraints.

One of the common approaches to effectively computing query answers in mediation systems is to reduce this problem into a query rewriting problem, usually called *query rewriting using*

[Hal01] A. Y. Halevy, "Answering queries using views: A survey", *VLDB Journal 10*, 4, 2001, p. 270–294.

[HIM+04] A. Y. Halevy, Z. G. Ives, J. Madhavan, P. Mork, D. Suciu, I. Tatarinov, "The Piazza Peer Data Management System.", *IEEE Trans. Knowl. Data Eng. 16*, 7, 2004, p. 787–798.

[AD98] S. Abiteboul, O. Duschka, "Complexity of Answering Queries Using Materialized Views.", *in: PODS*, p. 254–263, 1998.

views [Hal01,Len02,TH04]. Given a user query expressed over the global (or a peer) schema, the data sources that are relevant to answer the query are selected by means of a rewriting algorithm that allows to reformulate the user query into an equivalent or maximally subsumed (contained) query whose definition refers only to source descriptions.

The problem of rewriting queries in terms of views has been intensively investigated in the last decade (see [Hal01,Len02] for a survey). Existing research works differ w.r.t. the languages used to express a global schema, views and queries as well as w.r.t. the type of rewriting considered (i.e., maximally contained or equivalent rewriting). In a nutshell, this problem has been studied for different classes of languages ranging from various sub-languages of datalog, hybrid languages combining Horn rules and description logics to semistructured data models. Recently, the problem of rewriting queries in terms of views has been investigated in the context of P2P DBMSs [HIM+04,TH04].

4 Application Domains

As to the aspect dealing with flexible queries, there are several potential application domains. Soft querying turns out to be relevant in many contexts, such as information retrieval, in particular on the Web (many commercial systems, e.g. Google or Yahoo use a technique to rank-order the answers), yellow pages, classified advertisements, image or multimedia retrieval. One may guess that the richer the semantics of stored information (for instance images or video), the more difficult it is for the user to characterize his search criterion in a crisp way, i.e., using Boolean conditions. In this kind of situation, flexible queries which involve imprecise descriptions (or goals) and vague terms, may provide a convenient means for expressing information needs.

Even though most of the research works performed in Pilgrim assume relational data, many results can be transposed to other contexts such as information retrieval or multimedia database querying. We are currently working on the specification of a flexible route planning system involving fuzzy preferences (cf. Section 6.3), which should illustrate the utility of fuzzy queries in the context of intelligent transportation systems.

Databases involving imprecise data are not yet common in practice for two reasons: developing DBMSs supporting such data is a hard job and the demand is presently not so strong. However, many potential domains could take advantage of such advanced systems capable of storing and querying databases where some pieces of information are imprecise: military information systems, automated recognition of objects in images, data warehouses where information coming from more or less reliable sources must be fused and stored, etc.

[Hal01] A. Y. Halevy, "Answering queries using views: A survey", *VLDB Journal 10*, 4, 2001, p. 270–294.

[Len02] M. Lenzerini, "Data Integration : A Theoretical Perspective", Madison, Wisconsin, 2002.

[TH04] I. Tatarinov, A. Halevy, "Efficient query reformulation in peer data management systems", *in: SIGMOD '04: Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, ACM Press, p. 539–550, New York, NY, USA, 2004.

[HIM+04] A. Y. Halevy, Z. G. Ives, J. Madhavan, P. Mork, D. Suciu, I. Tatarinov, "The Piazza Peer Data Management System.", *IEEE Trans. Knowl. Data Eng. 16*, 7, 2004, p. 787–798.

5 Software

We are currently implementing a flexible querying prototype that aims at evaluating fuzzy queries addressed to regular databases. More precisely, it takes the form of an additional software layer on top of Oracle, whose function is to translate a fuzzy query into a procedural evaluation program including regular SQL queries in order to take advantage of the optimization mechanisms that exist in the DBMS. In its current version, the prototype, called FRIGA (Flexible RetrIeval and GRaded Answers), is able to process "simple" fuzzy queries (i.e., fuzzy queries involving a single block) and we are now extending it so as to make it support nested queries and so-called contextual fuzzy queries (i.e. queries where some fuzzy terms do not have to be explicitly defined by the user but whose interpretation depends on a certain context that can be determined from the query itself).

In 2008-09, we developed FlexIS, a flexible data mediation prototype using an LAV type of approach. This prototype aims at rewriting user queries in terms of views in a tolerant way, on the basis of the interval constraints involved in both the query and the views. The rewritings obtained are assigned a weight which reflects the probability that any tuple returned by that rewriting satisfies the constraints from the query. FlexIS can provide the user with either the best k rewritings of the query, or those whose degree is over a certain qualitative threshold. Some experimental results obtained with FlexIS are presented and analyzed in [23]. Both FRIGA and FlexIS should be made publicly available as Web services soon.

As mentioned in Section 4, we decided to illustrate the interest of flexible querying techniques in the domain of Intelligent Transportation Systems (ITS). Along with two other IRISA teams (namely Cairn and Cordial), Pilgrim is involved in the development of a platform called MOB-ITS in the context of the CPER INVENT'IST 2007-2013. This platform aims at supporting mobile and interactive access to information for ITS applications. In this framework, Pilgrim intends to implement an application that would make it possible for a mobile user to query distributed data sources according to a fuzzy variant of the "pay as you go" [SDH08] type of approach, i.e. without having available a global mediated schema of the sources that can potentially be queried. The flexible route planner outlined in Section 6.3 should also be integrated into that platform.

6 New Results

6.1 Possibilistic database modeling and querying

Participants: Patrick Bosc, Olivier Pivert.

Let us recall that an imprecise database corresponds to a set of interpretations (also called worlds) which are usual databases obtained by choosing a candidate value in each distribution. The combinatorial nature of this mechanism leads to a considerable number of worlds even when the number of ill-known values in the database is relatively small. A crucial objective is thus to define a "compact" processing method for algebraic queries, i.e., a method that does

[SDH08] A. D. Sarma, X. Dong, A. Halevy, "Bootstrapping pay-as-you-go data integration systems", *in: Proceedings of the ACM SIGMOD Conference*, p. 861–874, Vancouver (Canada), 2008.

not require to make the interpretations of the database explicit. Consequently, it is mandatory to design a data model that constitutes a strong representation system for the query language considered, i.e., a model that supports a closed set of operations whose results are consistent with a world-based interpretation. In other terms, if $rep(D)$ denotes the set of worlds associated with the imprecise database D, the following property must hold for any operation o of the language: $o(rep(D)) = rep(o(D))$. Such a framework guarantees a sound semantics for the operators of the language, and it makes it possible to envisage a tractable query evaluation process.

In the previous years, we defined a possibilistic database model involving usual possibilistic relations enriched with:

- a degree N associated with every tuple, which expresses the certainty that the tuple has a representative in any world than can be derived from the relation,
- nested relations used to represent dependencies between candidate values for different attributes and thus to model possibility distributions over several attributes,

and we showed that this model constitutes a strong representation system for the operations of selection, projection, union and foreign key join (fk-join). In 2009, we investigated the following three issues.

- Incomplete possibility distributions.

  In [18], we extended the aforementioned possibilistic database model so as to deal with incomplete possibility distributions. The situation considered is that where the data provider does not have a complete knowledge of the attribute domains involved and is only able to specify more or less possible and completely impossible candidates for some attribute values. For instance, let us consider a relation describing images of enemy aircrafts taken by spy satellites. Let us assume that every image represents a unique aircraft whose type may be ill-known due to the imprecision inherent in the automatic recognition process. One wants to take into account the possibility that the reference "catalogue" of airplane types used in the matching phase is incomplete (some aircraft types may be missing). Then, for a given image, some explicit types (from the catalogue) may be recognized as more or less possible, some others (also from the catalogue) may be classified as impossible, and there is a non-zero possibility that the aircraft in the picture matches an unknown type (absent from the catalogue). The model we proposed in [18] is based on the notion of an imprecise twofold value. Such a value is made of a positive part consisting of a weighted set (possibility distribution) which represents the more or less possible candidates, including — if necessary — a special value meaning "unknown", and a negative part consisting of a regular set which represents the totally impossible candidates. We have shown that this framework is a strong representation system for selection (with a restricted type of conditions), projection and union.

- Functional dependencies.

  In [17], we introduced a definition of the concept of a functional dependency (FD) in the context of databases containing ill-known attributes values represented by possibility distributions. Contrary to previous proposals, this definition is based on the possible worlds model and consists in viewing the satisfaction of an FD by a relation as an uncertain event whose possibility and necessity can be quantified. We gave the principle of a method for incrementally computing the related possibility and necessity degrees and tackled the issue of tuple refinement (data cleaning) in the presence of an FD.

- Certainty-based model.

  In [11], we dealt with the modeling and querying of a database containing uncertain attribute values, in the situation where some knowledge is available about the more or less *certain* value (or disjunction of values) that a given attribute in a given tuple can take. A relational database model suited to this context was introduced. As the possibilistic database model that we proposed previously, this "certainty-based" model is based on possibility theory, but it represents the values that are more or less certain instead of those which are more or less possible. This corresponds to the most important part of information (in this approach, a possibility distribution is "synthetized" by keeping its most plausible elements). The idea is to attach a certainty level to each piece of data (by default, a piece of data has certainty 1). Certainty is modeled as a lower bound of a necessity measure. For instance, $\langle$037, John, (40, α)$\rangle$ denotes the existence of a person named John, whose age is 40 with certainty α. Then the possibility that his age differs from 40 is upper bounded by $1 - \alpha$ without further information on the respective possibility degrees of other possible values. Selection, join and union operators of relational algebra were extended so as to handle relations in the framework of this certainty-based model. It was shown that i) the model in question is a strong representation system for the algebraic operators considered, and that ii) the data complexity associated with the extended operators is the same as in the classical database case, which makes the approach highly scalable. A possibilistic logic encoding of the model was also outlined.

6.2 Flexible querying of classical databases

6.2.1 Preference modeling

Participants: Olivier Pivert, Patrick Bosc, Grégory Smits, Amine Mokhtari.

- Contextual predicates.

  In [9, 10], we dealt with the interpretation and processing of database queries with preference conditions of the form "attribute is low (resp. medium, high)" in the situation where the user is not aware of the actual content of the database but still wants to retrieve the best possible answers (relatively to that content). As an example, let us consider a user who wants to move to a different country and aims at finding the "best" cities to settle down according to the following preferences: population between 50,000 and 100,000 (hard constraint), low average price of the square meter, low crime rate, medium average annual temperature. We assume that the user does not know the values taken by the attributes "average price of the square meter", "crime rate", "average annual temperature" in the country considered but still wants to find the best solutions relatively to the existing possibilities. The idea that we advocate is to use the fuzzy-set-based framework and to define the fuzzy predicates "high", "medium" and "low" in a relative way, using the minimal, average and maximal values of the attribute values present in the associated query-defined context (in the example above: the cities whose population is between 50,000 and 100,000). An approach to the definition of the terms "low", "medium" and "high" in a contextual and relative manner is introduced in [10], as well as an extension of the SQLf language allowing to express such contextual fuzzy conditions (possibly in a nested way). An algorithm aimed at efficiently retrieving the top-k answers to such queries is described in [9].

- Competitive conditional preferences.

  In [19], we introduced a new type of database queries involving fuzzy preferences. The starting point was hierarchical preference queries, whose basic form is: find the items satisfying C among which prefer those satisfying P_1, among which prefer those satisfying P_2 ... among which prefer those satisfying P_n. In [19], we extended this pattern so as to deal with queries where preferences are modelled as a tree (and not anymore as a list) of conditions. As in previously proposed models, the approach we advocate involves the notion of hierarchy (associated with conditional statements), but the novelty is that it also involves a notion of competition between the "children conditions" of a node, which correspond to a disjunction of *non mutually exclusive* predicates. Consider for instance the query: "find the persons who are preferably young (preference degree 1) or well paid (preference degree 0.8); if young then preferably tall (preference degree 1) or educated (preference degree 0.6); if well paid then preferably live in Paris (preference degree 1)". Since a person can be both (somewhat) young and (somewhat) well paid, these criteria "compete" at level 1, and the same phenomenon may occur at the other levels of the hierarchy. It is thus necessary to devise an interpretation that takes into account the fuzziness of the predicates, the weights attached to them, the hierarchical aspect linked to conditional statements, and the competition between the preference conditions. In [19], two possible interpretations of such queries are defined and two associated evaluation techniques are outlined.

- Outranking and classification.

  In [12], we presented a (non-fuzzy) approach to database flexible querying inspired by the concept of outranking (which corresponds to a weighted majority rule) in a decision-making context, as an alternative to Pareto-ordering. The method assumes available a set of scoring functions which translate incommensurable partial preferences (let us recall that fuzzy-set-based approaches only handle commensurable ones). The proposal relies on an outranking measure which aggregates the numbers of partial preferences either concordant, discordant or indifferent with a given ordering between a tuple and the profile describing a given class. Indeed, instead of being compared pairwise like in Pareto-order-based approaches, tuples are compared to acceptability profiles that are associated with user-defined classes. According to its satisfaction of the partial preferences involved in the query, a tuple is assigned to a given class with a certain degree. An algorithm which performs that classification with a linear data complexity is described in [12].

6.2.2 Extended division operators

Participants: Patrick Bosc, Olivier Pivert.

The role and properties of the division are well-known in the context of queries addressed to regular relational databases. Basically, it is intended to retrieve the values X (of the dividend built over X and A) which are associated with all of the elements A of the divisor. This operator can be extended in several directions when preferences come into play. In particular, it may apply to fuzzy or ordinal relations and some tolerance may be taken into account. These issues have been investigated along with a special attention to a layered divisor and the relationship between division and a derived operator called antidivision on the one hand and information retrieval on the other hand. In all the works undertaken, a constant objective is to define division operators which deliver a result that can be characterized as a quotient. In other words, the relation returned is maximal in terms of the inclusion of its product with the divisor in the dividend.

- Tolerant division.

  In [3], some tolerance is introduced in the division so as to cope with either quantitative or qualitative exceptions. In the first case, ones looks for the values X which are associated with *almost all* the values of the divisor; in other words, some missing or low satisfactory associations may be more or less ignored depending on the modeling of the weakened quantifier "almost all". According to the second view, we want to deal with situations close to full satisfaction, but leading to a low satisfaction degree. In such a case called a low-intensity exception, an upgrade takes place and the tolerant division expresses that the divisor is *almost included* in the set of elements of the dividend related to the the considered X-value. It is shown that the result delivered is a quotient in both cases.

- Stratified division.

  Another type of extended division, called a stratified division, is defined when the divisor is specified no longer as a flat set, but as several subsets of elements. These subsets are used in a hierarchical way in the context of the associations related to X-values. One of the interests of such an approach is to get rid of membership functions and numeric grades since the elements of the resulting relations may be qualified in a purely qualitative way. A first vision [15] consists of queries of the form: retrieve the best k values which are associated with *set-1* and if possible with *set-2* ... and if possible with *set-n.* This approach can be seen as a refinement of the regular division where the first set is mandatory and the others serve for breaking ties. Two other types of use of a stratified divisor are suggested in [14]. The first one is based on a disjunctive interpretation of the divisor leading to queries asking for the best k values which are associated with *set-1* or else with *set-2* ... or else with *set-n.* A third interpretation is proposed where all the layers intervene in the determination of the level of satisfaction. According to this view, the satisfactions of the various layers give birth to a binary vector and the higher the value of the corresponding number, the more satisfactory the considered X-value. The three types of resulting relations may be characterized as quotients. Moreover, we have investigated the issue of implementing such queries. Diverse strategies have been tested and the conclusion is that good performances can be expected only with an implementation in the core of the DBMS.

- Stratified antidivision.

  The antidivision is similar to the division except that the elements of interest are those which are connected with none of the values of the divisor. A typical example is the search of components which contain no elements of a given list (which plays the role of an “antidivisor”). We have studied the counterpart of the stratified division for the antidivision [13]. The three previous types of queries make sense and their implementation is feasible in a similar way and comparable performances are obtained.

- Application to IR.

  The connection existing between the division operation and information retrieval has been recognized for a long time. The basic reason of the relationship lies in the fact that both mechanisms are founded on the notion of inclusion. In [8, 20, 33, 34], it is shown how to choose appropriate tools (mainly fuzzy implications and conjunctions) in order to obtain performances in terms of recall and precision which are at the level of the reference system OKAPI-BM25 (and sometimes sligthly better).

6.2.3 Bipolar fuzzy queries

Participants: Patrick Bosc, Ludovic Liétard, Olivier Pivert, Daniel Rocacher.

Bipolar queries provide a way to integrate preferences inside queries where mandatory preferences, called constraints, are distinguished from optional ones, called wishes. A constraint (resp. a wish) is defined by a set of acceptable (resp. desired) values. Tuples satisfying the constraints and the wishes are returned in priority to the user. Then, tuples satisfying only the constraints are delivered. We consider the case of bipolar conditions where both the wish and the constraint are defined by fuzzy sets thus defining bipolar fuzzy queries [27].

- Aggregation of fuzzy bipolar conditions.

  This year, we have pointed out the importance of defining an extended norm and an extended co-norm for aggregating fuzzy bipolar conditions. We have proposed in [26] a couple of extended norm and co-norm which constitutes a generalization of the couple (*min*, *max*) used in the fuzzy set framework. In addition, we have proposed other connectives suited to bipolar fuzzy predicates, which have no counterpart in a classical (i.e., non-bipolar) fuzzy set context. However, in the particular case where fuzzy bipolar conditions are fuzzy conditions, some of these operators lead to a refinement of norm *min* and conorm *max*. Furthermore, the algebraic operators of selection, join and projection have been extended [25] so as to deal with bipolar conditions and/or relations.

- Bipolar division.

  In [16], we investigated how bipolarity may impact the division operator in the context of relational databases. Various forms of bipolar divisions can indeed be devised, each of them conveying a specific semantics. Starting with a basic bipolar division with crisp relations where the divisor is made of two components (one representing values which are required, the other describing the values which are expected but not mandatory), we moved to more sophisticated forms of bipolar divisions: i) the dividend is a graded (fuzzy) relation where each tuple has a degree of membership to the fuzzy concept conveyed by the relation and ii) the universal quantifier is softened into possibly two different weaker forms (one related to the constraint, i.e., the mandatory values of the divisor, and the other for the wish part, i.e., the values of the divisor that are only desired). The result of all these bipolar divisions is characterized as a quotient, i.e., a maximal relation.

6.2.4 Cooperative answering to flexible database queries

Participants: Allel Hadjali, Olivier Pivert, Hélène Jaudoin, Patrick Bosc.

Even though the use of fuzzy queries reduces the risk of obtaining empty answers, this situation can still occur when no element of the database satisfies the query even at a low degree. In 2009, we have proposed two approaches for dealing with failing fuzzy queries.

- Incremental relaxation based on fuzzy set dilation.

  The approach proposed in [2] only exploits the user query that results into an empty set of answers. It relies on an appropriate transformation that allows for enlarging gradual predicates involved in a failing query, by means of a dilation operation. A convenient parameterized proximity relation is used as a basis for that operation. The resulting predicates are semantically close to the initial ones and their computation boils down to simple fuzzy arithmetic operations. A rigorous controlling tool for query relaxation is discussed as well. Moreover, we have shown how conjunctive fuzzy queries can be relaxed locally, i.e., only some subqueries are affected by the relaxation process. To make the search for a non-failing relaxed query more efficient and to avoid a brute search procedure (which is time consuming), a technique which exploits the MFSs (Minimal Failing Subqueries) of the original query is used. This technique allows for pruning the search space and results into a substantial performance improvement.

- Semantic proximity between predicates.

  This second approach [7, 21] consists in replacing a failing query Q by another which i) has been processed previously, ii) is semantically similar to Q, iii) returns a non-empty set of answers. It is assumed that the system stores the non-failing queries in a repository, and that a fuzzy resemblance measure over every attribute domain involved in the database is available. The key point in this approach is to define the notion of semantic proximity between queries, or rather of semantic *substitutivity* since the objective is to replace a failing query Q_1 by a non-failing query Q_2 but not the opposite. This method avoids the combinatory explosion induced by the relaxation of the predicates from a conjunctive query. Indeed, there exists in general a high number of relaxed queries and one cannot know whether these queries provide a non-empty answer before processing them. With the approach proposed, when a substitute exists in the repository, only one query needs to be processed, with the guarantee of obtaining a non-empty answer “in one shot”.

6.2.5 Gradual numbers

Participants: Daniel Rocacher, Patrick Bosc, Ludovic Liétard.

This work covers the following two aspects, which both concern the application of gradual numbers to database flexible querying:

- Fuzzy relative integers and fuzzy bags

  A characterization of fuzzy bags with conjunctive fuzzy integers ($\mathbb{N}_f$) provides a general framework in which sets, bags, fuzzy sets and fuzzy bags are treated in a uniform way. In this context, the difference between two fuzzy bags does not always exist and, in such cases, only approximations of the difference have been defined. The problem comes from the fact that the fuzzy bag model considered so far is based on positive fuzzy integers. In consequence, we have constructed [6] the set of fuzzy relative integers ($\mathbb{Z}_f$) where the existence of negative multiplicities allows to define exact complementations. This new concept offers a well-founded framework in which the difference of two fuzzy bags is always defined. This approach has interesting applications in flexible querying of databases but it has also a larger scope and can be applied to many domains such as fuzzy data mining, summarization, or fuzzy information retrieval.

- Conditions with aggregates

  Gradual number theory provides a new framework to define a fuzzy quantity as a collection of more or less convenient interpretations. In this context, mathematical operations such as the difference, the division, the minimum and the maximum can be exactly evaluated. We have shown [5] that these operations can be used to define complex fuzzy conditions involving a mathematical expression computed on a fuzzy referential. It becomes possible to evaluate quantified statements and conditions of type "$agg(A)$ is C" where A is a fuzzy multiset, C a fuzzy predicate and agg is either the aggregate average, minimum or maximum (as in "the average salary of *young* employees is *high*"). Such an evaluation provides a gradual truth value which represents the different interpretations of the result, each interpretation being an exact evaluation of the predicate "the aggregate satisfies C" computed on an α-cut. A defuzzification process can be applied on the gradual truth value in order to obtain a scalar result which can be viewed as a kind of summary (either qualitative or quantitative).

6.3 Personalized route planning involving fuzzy preferences

Participants: Olivier Pivert, Amine Mokhtari, Allel Hadjali.

In 2008, we started investigating the application of fuzzy set theory to flexible route planning. In 2009, the results obtained concerned the following four issues:

- Typology of preferences.

  Route planners are systems which help users select a route between two locations. In such a context, personalization mechanisms notably aim at taking into account user preferences so as to identify the best route(s) among a set of possible answers. In this field, the nature of potential user preferences presents a great variety. It is then important to characterize them and classify them in order to make their handling easy and efficient. In [29], we have provided a typology of preferences which make sense in the context of unimodal point-to-point route planning. Three families of user preferences are distinguished: i) spatial preferences; ii) global preferences; iii) spatio-temporal preferences. The first family aims at capturing the wish of a user to pass or not by particular roads, places or some parts of a road network. For example, "avoid secondary roads". The second one concerns some basic properties of a route seen as a whole, such as comfort, length, duration or safety. For example, "prefer a fast route". The last one gathers preferences which are spatial (resp. global) preferences involving a time component whose purpose is to express the moment or period when the spatial (resp. global) preference is relevant. An example is "avoid the city center around noon". In [29], we showed that gradual predicates modelled by fuzzy sets represent a convenient way for expressing such preferences.

- Fuzzy query language loosely based on SQL.

  In [22], we have proposed a first representation, loosely based on SQL, for a unimodal point-to-point route planning query. The representation proposed involves the following basic components: i) the departure (resp. arrival) parameter that contains information about the place and time of departure (resp. arrival), ii) a compound preference that involves a set of atomic user preferences connected by a fuzzy operator (conjunction, mean, etc). Two query semantics are considered: static and dynamic. Static queries consist in computing the relevant routes between two places s_a and s_b for a set of preferences P at a given instant of departure t_d (resp. arrival t_a). Such a query involves either t_d or t_a or none of them, but not both. On the other hand, in dynamic queries the parameter t_d (resp. t_a) is an interval. Such a query specifies either t_a or t_d. The objective is not only to return the relevant routes, but also to determine the best moment for the trip.

- Bipolar fuzzy query language based on tuple relational calculus.

  In [30, 31], we proposed a refined version of the query language, which relies on tuple relational calculus. In this framework, a route planning query Q can be expressed as follows:

  $$Q = [\Omega/COND,\ k],$$

  where Ω is an element of the relation *Route* (which stores the shortest routes linking the two locations of interest and is initialized by a pre-processing step) and *COND* is a formula that contains the set of preferences P and a condition about the departure (resp. arrival) place. As to k, it denotes the desired number of answers in the result (the best, in accordance with the top-k query paradigm). Considering that preferences in a route planning context are often of a bipolar nature, the set of preferences P is partitioned into two sets P_C and P_W. The former (resp. the latter) represents the set of conditions that the user views as (possibly flexible) constraints (resp. wishes). Fuzzy constraints correspond to "negative" preferences in the sense that their complements define fuzzy sets of values that are rejected as being non-acceptable. In [30, 31], the issue of defining the semantics of the fuzzy predicates which model preferences was also investigated. Two kinds of predicates were identified: i) user-defined predicates (where the semantics is explicitly given by the user) as in "route.cost is *high*"; ii) automatically defined predicates (where the semantics is computed by the system according to some contextual information) for instance "route.duration is *comparatively short*".

- Evaluation strategy and system architecture.

  In [28], we defined on the one hand the basic architecture of a DBMS supporting route planning queries, and on the other hand the query processing strategy. The system we propose consists of four modules: a parser, an optimizer, a path generator and an evaluator. The first one checks the correctness of the query at a syntactical level. The optimizer generates an efficient query plan, while the purpose of the path generator is i) to compute the set of k' ($> k$) shortest routes between the two locations of interest in the graph representing the road network, and ii) to initialize the relations *Route* and *Segment* mentioned above. As for the evaluator, it proceeds in three steps:

  1. it builds the membership functions of the automatically defined predicates involved in a user query Q (if any). The relations *Route* and *Segment* are used as the context of these predicates ;
  2. it evaluates each route $\Omega \in Route$ w.r.t. each atomic user preference present in P;
  3. it aggregates the degrees into a single score (or two: one for the constraint, one for the wish, depending on how bipolarity is handled), sorts the tuples from *Route* according to this (or these) score(s) and returns the top k answers.

We are now working towards modelling the *user context* in order to enrich the set of preferences involved in a route planning query.

6.4 Flexibility issues in data integration systems

Participants: Hélène Jaudoin, Olivier Pivert, Allel Hadjali.

These recent years, we have been tackling the issue of adding some flexibility to data integration systems. In 2009, three distinct lines have been investigated.

6.4.1 Value constraints

In the context of data integration in open environments such as the web, it is important to reduce the number of useless query rewritings. The more accurate the description of views, the smaller the number of rewritings. In order to enable fine-grained description of views, it is interesting to consider value constraints on attributes, i.e., enumeration of possible/authorized values of the attributes, in the description of views and queries. Those constraints allow for specifying queries of the form: "retrieve the individuals whose values on a given attribute cannot be outside of the set of values $\{a_1, ..., a_n\}$". In [4], a sound and complete query rewriting algorithm, i.e., that computes all certain answers to a given query, in the presence of value constraints on attributes has been proposed. The algorithm is based on data mining techniques and a hypergraph framework in order to reach a good scalability of the implementation.

6.4.2 Tolerant rewriting

In a context where data sources are autonomous, integration systems are confronted to the problem of imperfect matchings between the value domains of the views and those involved in the queries. Indeed, it is not realistic in general to assume finding views which totally satisfy the domain constraints imposed by the query and which are able to provide certain answers. As an example, let us consider a query Q that asks for *names* of *persons* whose *age* is in the interval $[28, 38]$ and two views V_1 and V_2 such that:
V_1 provides *names* of *persons* whose *age* is in $[25, 35]$ and
V_2 provides *names* of *persons* whose *age* is in $[36, 46]$.
V_1 and V_2 both have an interval constraint on the attribute *age* but none of these intervals is included in that of the query. Moreover, as V_1 and V_2 only return names of persons, a selection on attribute *age* is impossible, and consequently, V_1 and V_2 cannot provide any certain answer to Q. Contrary to what is done by regular query rewriting algorithms, we propose not to discard such views since they can still return a significant number of correct answers. In [24, 23], we focused on the problem of computing "probable answers" to a query and we defined the notion of a tolerant rewriting which allows for finding such answers. Each tolerant rewriting Q' of a query Q is attached a degree that corresponds to the probability for a tuple returned by Q' to be a correct answer to Q. An issue is that the number of possible rewritings with respect to this semantics can be huge. Therefore, we exploit the degrees so as to rank-order the rewritings: in [24, 23] an algorithm is proposed which directly computes the *best* rewritings in decreasing order of their degrees.

6.4.3 Mapping generation

A fundamental problem raised by the creation of a data integration system lies in the discovery of mappings between data sources schemas. In order to enable the combination of data supplied by autnomous data sources in the setting of peer-to-peer data integration systems, the generation of *decentralized* semantic mappings, i.e., semantic mappings between the different source schemas becomes necessary. Such mappings are query expressions such as $Q(S_1) \sqsubset Q'(S_2)$ (resp. $Q'(S_2) \sqsubset Q(S_1)$) where S_1 and S_2 are schemas of two different sources. In [32], we dealt with the problem of automatic generation of decentralized mappings between data sources schemas, starting from mappings between these schemas and a global ontology. This problem is formalized in the framework of description logics and boils down to a problem of query rewriting using views. Two subproblems were identified: the first one is equivalent to a classical problem of query rewriting using views whereas the second one consists in computing expressions that *generalize* a query on a given schema instead of specializing it. In the case of description logics with the property of structural subsumption, we have shown that this latter problem boils down to the computation of minimal transversals of a hypergraph.

7 Other Grants and Activities

7.1 National actions

Patrick Bosc, Allel Hadjali, Hélène Jaudoin and Olivier Pivert participated in the ANR project “FORUM”, which ended in June 2009 and which dealt with the problem of information integration in a large and highly dynamic information space. The other teams involved were from LIRMM (Montpellier), LIMOS (Clermont-Ferrand), LIRIS (Lyon), and CEMAGREF (Clermont-Ferrand).

Ludovic Liétard, Allel Hadjali, and Daniel Rocacher participate in the ANR project “AOC”, which deals with the definition of matching methods for complex objects (graphs in particular). The other teams involved are from IRIT (Toulouse), PRISM (Versailles), LIRIS (Lyon).

7.2 International actions

Leonid Tineo, from University Simon Bolivar (Caracas, Venezuela) spent a sabbatical year (from Jan. 2009 to Dec. 2009) in the team Pilgrim.

Carmen Brando, from University Simon Bolivar (Caracas, Venezuela) did her Master’s degree internship in our team in 2009. Her research topic was about the definition of an approach based on query reuse for handling failing fuzzy queries.

8 Dissemination

8.1 Teaching

Project members give lectures in different faculties of engineering, in the third cycle University curriculum: ”Bases de données, gradualité et imprécision“ in the speciality ”Intelligence Artificielle et Images“ of the Master’s degree in computer science at University of Rennes 1, and at Enssat (third year level cursus).

A. Hadjali gave a Master’s course entitled "Requêtes à préférences" at ESI (Ecole Supérieure d’Informatique), University of Algiers, 2009.

8.2 Scientific activities

8.2.1 Invited talks

O. Pivert gave an invited talk on possibilistic databases at the VLDB Workshop on the Management of Uncertain Data (MUD’09).

8.2.2 Program committees

P. Bosc served as a member of the following program committees:

- 24^{th} ACM Symposium on Applied Computing, Special Track on Information Access and Retrieval, Honolulu, Hawaii, USA, March 8–12, 2009.

- 8th International Conference on Flexible Query-Answering Systems (FQAS'09), Roskilde (Denmark), October 26–28, 2009.
- 9emes Journées Francophones Extraction et Gestion des Connaissances (EGC'09), Strasbourg, France, 27–30 janvier 2009.
- Rencontres Francophones sur la Logique Floue et ses Applications (LFA'09), Annecy, France, 5–6 novembre 2009.
- DEXA 2009 4th International Workshop on Flexible Database and Information Systems Technology (FlexDBIST'09), Linz, Austria, August 31–September 4, 2009.
- 1^{ere} Conférence Francophone sur les Technologies de l'Information, de la Communication et de la Géolocalisation dans les Systèmes de Transport (CoGIST'09), Saint-Quay-Portrieux, 29 juin–1er juillet, 2009.
- IEEE/WIC International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI'09/IAT'09), Milan, Italy, September 15–18, 2009.
- WI'09/IAT'09 Workshop on Soft approaches to Information Access on the Web, Milan, Italy, September 15–18, 2009.
- VLDB 2009 3rd International Workshop on the Management of Uncertain Data (MUD'09), Lyon, France, August 28, 2009.
- Joint IFSA World Congress (IFSA'09) and the 6th Conference of the European Society of Fuzzy Systems and Technology (EUSFLAT'09), Lisbon, Portugal, July 20-24, 2009.
- 28th International Conference of the North American Fuzzy Information Processing Society (NAFIPS'09), Cincinnati, Ohio, USA, 2009.
- 20th Int. Conference on Database and Expert Systems Applications (DEXA'09), Linz, Austria, August 31-September 4, 2009.
- 2nd North American Simulation Technology Conference (NASTEC 2009), Atlanta, USA, August 26–28, 2009.

A. Hadjali served as a member of the following program committees:

- Rencontres Francophones sur la Logique Floue et ses Applications (LFA'09), Annecy (France), 5–6 novembre 2009.
- 3emes Journées Francophones sur les Ontologies (JFO'09), 3–4 décembre 2009, Poitiers.
- 8th International Conference on Flexible Query-Answering Systems (FQAS'09), Roskilde (Denmark), October 26–28, 2009.
- 18th International Symposium on Methodologies for Intelligent Systems (ISMIS'09), Prague (Czech Republic), September 14–17, 2009.

- Joint IFSA World Congress (IFSA'09) and the 6^{th} Conference of the European Society of Fuzzy Systems and Technology (EUSFLAT'09), Lisbon, Portugal, July 20-24, 2009.
- 6^{eme} Colloque sur l'Optimisation et les Systèmes d'Information (COSI'09), Annaba, Algérie, 25-27 mai, 2009.

O. Pivert served as a member of the following program committees:

- 24^{th} ACM Symposium on Applied Computing, Special Track on Information Access and Retrieval, Honolulu, Hawaii, USA, March 8–12, 2009.
- 8^{th} International Conference on Flexible Query-Answering Systems (FQAS'09), Roskilde (Denmark), October 26–28, 2009.
- Rencontres Francophones sur la Logique Floue et ses Applications (LFA'09), Annecy (France), 5–6 novembre 2009.
- DEXA 2009 4^{th} International Workshop on Flexible Database and Information Systems Technology (FlexDBIST'09), Linz, Austria, August 31–September 4, 2009.
- 3^{rd} International Scalable Uncertainty Management Conference (SUM'09), Washington DC, USA, September 28–30, 2009.
- 1^{ere} Conférence Francophone sur les Technologies de l'Information, de la Communication et de la Géolocalisation dans les Systèmes de Transport (CoGIST'09), Saint-Quay-Portrieux, 29 juin–1er juillet, 2009.
- 25^{emes} Journées Bases de Données Avancées (BDA'09), Namur, Belgique, 20–23 octobre 2009.
- WI'09/IAT'09 Workshop on Soft approaches to Information Access on the Web, Milan, Italy, September 15–18, 2009.
- VLDB 2009 3^{rd} International Workshop on the Management of Uncertain Data (MUD'09), Lyon, France, August 28, 2009.

D. Rocacher served as a member of the following program committee:

- Rencontres Francophones sur la Logique Floue et ses Applications (LFA'09), Annecy (France), 5–6 novembre 2009.

8.2.3 Organizing committees

Patrick Bosc, Allel Hadjali and Olivier Pivert co-organized a special session entitled "Advances in Soft Computing Applied to Database and Information Systems" at IFSA-EUSFLAT Conference, Lisbon, Portugal, July 20-24, 2009.

Patrick Bosc chaired the organizing committee of the first French-speaking conference on Information Technologies, Communication and Geolocation in Transportation Systems (CoGIST'09), which took place in Saint-Quay-Portrieux, June 29 – July 1, 2009.

8.2.4 Editorial boards

Patrick Bosc is a member of the following editorial boards:

- International Journal on Fuzziness, Uncertainty and Knowledge-Based Systems,
- Fuzzy Sets and Systems,
- Revue I3.

9 Bibliography

Major publications by the team in recent years

[1] P. Bosc, L. Liétard, O. Pivert, D. Rocacher, *Gradualité et imprécision dans les bases de données*, Ellipses, 2004.

[2] P.Bosc, O. Pivert, D.Rocacher, "About quotient and division of crisp and fuzzy relations", *Journal of Intelligent Information Systems 29*, 2, 2007, p. 185–210.

[3] P.Bosc, O. Pivert, "About projection-selection-join queries addressed to possibilistic relational databases", *IEEE Transactions on Fuzzy Systems 13*, 1, 2005, p. 124–139.

[4] P.Bosc, O. Pivert, "About possibilistic queries and their evaluation", *IEEE Transactions on Fuzzy Systems 15*, 1, 2007, p. 439–452.

Books and Monographs

[1] P. Bosc, A. Hadjali, G. Pasi (editors), *Journal of Intelligent Information Systems, Special Issue on Flexible queries in information systems*, *33*, 3, 2009, 235–237p.

Articles in referred journals and book chapters

[2] P. Bosc, A. Hadjali, O. Pivert, "Incremental controlled relaxation of failing flexible queries", *Journal of Intelligent Information Systems 33*, 3, 2009, p. 261–283.

[3] P. Bosc, O. Pivert, D. Rocacher, "Tolerant division queries and possibilistic database querying", *Fuzzy Sets and Systems 160*, 15, 2009, p. 2120–2140.

[4] H. Jaudoin, F. Flouvat, J.-M. Petit, F. Toumani, "Towards a scalable query rewriting algorithm in presence of value constraints", *Journal on Data Semantics 12*, 2009, p. 37–65.

[5] L. Liétard, D. Rocacher, "Conditions with aggregates evaluated using gradual numbers", *Control and Cybernetics 38*, 2, 2009, p. 395–417.

[6] D. Rocacher, P. Bosc, "The set of fuzzy relative integers and fuzzy bags", *Int. J. Intell. Syst. 24*, 6, 2009, p. 677–696.

Publications in Conferences and Workshops

[7] P. Bosc, C. Brando, A. Hadjali, H. Jaudoin, O. Pivert, "Semantic proximity between queries and the empty answer problem", *in: Proc. of the Joint IFSA World Congress (IFSA'09) and the 6th Conference of the European Society of Fuzzy Systems and Technology (EUSFLAT'09)*, p. 259–264, Lisbon, Portugal, 2009.

[8] P. Bosc, V. Claveau, O. Pivert, L. Ughetto, "Graded-inclusion-based information retrieval systems", *in: Proc. of the 31st European Conference on Information Retrieval (ECIR'09)*, p. 252–263, Toulouse, France, 2009.

[9] P. Bosc, O. Pivert, A. Mokhtari, "On fuzzy queries with contextual predicates", *in: Proc. of the 18th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'09)*, p. 484–489, Jeju, South Korea, 2009.

[10] P. Bosc, O. Pivert, A. Mokhtari, "Top-k queries with contextual fuzzy preferences", *in: Proc. of the 20th Int. Conference on Database and Expert Systems Applications (DEXA'09), LNCS 5690*, p. 846–853, Linz, Austria, 2009.

[11] P. Bosc, O. Pivert, H. Prade, "A model based on possibilistic certainty levels for incomplete databases", *in: Proc. of the 3rd International Conference on Scalable Uncertainty Management (SUM'09)*, p. 80–94, Washington DC, USA, 2009.

[12] P. Bosc, O. Pivert, G. Smits, "A flexible querying approach based on outranking and classification", *in: Proc. of the 8th International Conference on Flexible Query Answering Systems (FQAS'09)*, p. 1–12, Roskilde, Denmark, 2009.

[13] P. Bosc, O. Pivert, O. Soufflet, "Anti-division queries with ordinal layered preferences", *in: Proc. of the 10th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'09), LNAI 5590*, p. 769–780, Verona, Italy, 2009.

[14] P. Bosc, O. Pivert, O. Soufflet, "On three classes of division queries involving ordinal preferences", *in: Proc. of the 18th International Symposium on Methodologies for Intelligent Systems (ISMIS'09), LNAI 5722*, p. 311–320, Prague, Czech Republic, 2009.

[15] P. Bosc, O. Pivert, O. Soufflet, "Stratified division queries involving ordinal user preferences", *in: Proc. of the 24th Annual ACM Symposium on Applied Computing (ACM SAC'09)*, p. 1748–1749, Honolulu, Hawaii, USA, 2009.

[16] P. Bosc, O. Pivert, "About bipolar division operators", *in: Proc. of the 8th International Conference on Flexible Query Answering Systems (FQAS'09)*, p. 572–582, Roskilde, Denmark, 2009.

[17] P. Bosc, O. Pivert, "Functional dependencies over possibilistic databases: An interpretation based on the possible worlds semantics", *in: Proc. of the 3rd Workshop on the Management of Uncertain Data (MUD'09), in conjunction with VLDB'09*, p. 1–15, Lyon, France, 2009.

[18] P. Bosc, O. Pivert, "On a possibilistic database model with incomplete possibility distributions", *in: Proc. of the 28th International Conference of the North American Fuzzy Information Processing Society (NAFIPS'09)*, p. 1–6, Cincinnati, Ohio, USA, 2009.

[19] P. Bosc, O. Pivert, "On two interpretations of competitive conditional fuzzy preferences", *in: Proc. of the 1st International Conference on Fuzzy Computation (ICFC'09)*, p. 71–74, Madeira, Portugal, 2009.

[20] P. Bosc, L. Ughetto, O. Pivert, V. Claveau, “Implication-based and cardinality-based inclusions in information retrieval”, *in: Proc. of the 18th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE’09)*, p. 2088–2093, Jeju, South Korea, 2009.

[21] A. Brikci-Nigassa, A. Hadjali, “Finding the best relaxations of database queries in a flexible setting”, *in: Actes du 6ème Colloque sur l’Optimisation et des Systèmes d’Information (COSI’09)*, Annaba, Algérie, 2009.

[22] A. Hadjali, A. Mokhtari, O. Pivert, “Integrating complex user preferences into a route planner: A fuzzy-set-based approach”, *in: Proc. of the Joint IFSA World Congress (IFSA’09) and the 6th Conference of the European Society of Fuzzy Systems and Technology (EUSFLAT’09)*, p. 501–506, Lisbon, Portugal, 2009.

[23] H. Jaudoin, P. Colomb, O. Pivert, “Ranking approximate query rewritings based on views”, *in: Proc. of the 8th International Conference on Flexible Query Answering Systems (FQAS’09)*, p. 13–24, Roskilde, Denmark, 2009.

[24] H. Jaudoin, P. Colomb, O. Pivert, “Réécriture tolérante de requêtes en termes de vues”, *in: Actes des 24èmes Journées Bases de Données Avancées (BDA’09)*, Namur, Belgium, 2009.

[25] L. Liétard, D. Rocacher, P. Bosc, “On the extension of SQL to fuzzy bipolar conditions”, *in: Proc. of the 28th International Conference of the North American Fuzzy Information Processing Society (NAFIPS’09)*, Cincinnati, Ohio, USA, 2009.

[26] L. Liétard, D. Rocacher, “On the definition of extended norms and co-norms to aggregate fuzzy bipolar conditions”, *in: Proc. of the Joint IFSA World Congress (IFSA’09) and the 6th Conference of the European Society of Fuzzy Systems and Technology (EUSFLAT’09)*, p. 513–518, Lisbon, Portugal, 2009.

[27] L. Liétard, D. Rocacher, “Requêtes à préférences et bipolarité”, *in: Actes du 15ème Colloque National de la Recherche en IUT (CNRIUT)*, Villeneuve d’Ascq, 2009.

[28] A. Mokhtari, O. Pivert, A. Hadjali, P. Bosc, “Towards a route planner capable of dealing with complex bipolar preferences”, *in: Proc. of the 12th IEEE Conference on Intelligent Transportation Systems (IEEE ITSC’09)*, p. 556–561, St. Louis, MO, USA, 2009.

[29] A. Mokhtari, O. Pivert, A. Hadjali, “An approach to personalized route planning based on fuzzy sets”, *in: Proc. of the International Scientific Conference on Mobility and Transport (mobil.TUM 2009)*, Munich, Germany, 2009.

[30] A. Mokhtari, O. Pivert, A. Hadjali, “Vers un langage de requête flou pour la planification d’itinéraire”, *in: Actes des Rencontres Francophones sur la Logique Floue et ses Applications (LFA 2009)*, p. 33–40, Annecy, France, 2009.

[31] A. Mokhtari, O. Pivert, A. Hadjali, “Vers un système de planification d’itinéraire intégrant des préférences complexes”, *in: Actes de l’atelier PeCUSI 2009 conjoint à INFORSID 2009*, p. 39–50, Toulouse, France, 2009.

[32] K. Toumani, H. Jaudoin, M. Schneider, “Automatic generation of P2P mappings between sources schemas”, *in: Proc. of the 18th International Symposium on Methodologies for Intelligent Systems (ISMIS’09), LNAI 5722*, p. 139–150, Prague, Czech Republic, 2009.

[33] L. Ughetto, O. Pivert, V. Claveau, P. Bosc, “Recherche d’information et inclusions graduelles”, *in: Actes des Rencontres Francophones sur la Logique Floue et ses Applications (LFA 2009)*, p. 125–132, Annecy, France, 2009.

[34] L. Ughetto, O. Pivert, V. Claveau, P. Bosc, “Système de recherche d’information à base d’inclusion graduelle”, *in: Actes de la 6ème Conférence en Recherche d’Information et Applications (CORIA’09)*, p. 235–250, Presqu’île de Giens, France, 2009.