Jornada de Seguimiento de Proyectos, 2005
Programa Nacional de Tecnologías Informáticas

Rules as a tool for the design and management of relational databases: algebraic foundations, deduction methods and efficient tools. TIC2003-08687-C02

Inma P. de Guzmán *
Univ. Málaga

Olga Pons †
Univ. Granada

Summary

Data connections are becoming an outstanding area in databases, involving dependencies (of several kinds) and associations. These data relations may be conceive as integrity rules which provide information about the database structure and, therefore, may help the design and management tasks. In this proposal, we intend to give a uniform and efficient solution to the problem of how to specify and to manipulate these data relations.

Normally, the work in this area is centered on the design of methods to discover new data relations directly from the database itself (for example, using data mining techniques). Another direction is to provide an uniform treatment of the different kinds of rules and to develop deduction methods to discover new rules from existing rules. Both approaches are complementary, because inferred rules allow us to prune the search in data mining techniques and rules discovered with these data mining methods provide new rules to the deduction method to carry out new inference tasks.

We propose the use of Logic to represent several types of data dependencies (concretely, we will focus on functional dependencies, multivalued dependencies, inclusion dependencies, fuzzy dependencies and temporal dependencies) and we will develop efficient deduction methods to manage these database rules. This approach allows a uniform and formal way to represent different levels of knowledge about the information system data and to prepare them to be manipulated in an automated way. The use of logic opens the doors to develop automated deduction methods which can be extended at the same time the research development is increasing. In this way, the theoretical result can be incorporated to the tools generated with lower cost.

*Email: guzman@ctima.uma.es
†Email: opc@decsai.ugr.es

1 Objectives of the project

The main aim of the project is to develop a general framework that allows us the specification and manipulation of database constraint, more concretely rules and dependencies. In the first stage of the project we have focussed on the development of a formal basis which allows to integrate different result in a efficient way.

The aim of the first sub-project is to develop deduction methods for Functional Dependencies logics. The kinds of logics may be used to design relational databases. Thus, the result have been applied to the field of Intelligent Information System, concretely, some CASE tools have been developed. These tools are based on a formal basis and they use the automated deduction techniques developed in this proposal to manage the database constraint in an efficient way.

The objectives in this sub-project are the following:

- From the starting point of the Functional Dependencies methateoretical results concerning Funtional Dependencies Logics developed by this subgroup in the past, we design efficient algorithms directly based on the logic deduction system. These algorithm face on with the most outstanding problems in this area: Canonical and Minimal Clousure, Normalization algorithms, the Key problem, etc.
- Extension of Functional Dependencies logic to manage other kinds of dependencies (Fuzzy, Temporal, etc). This objective looks for a uniform way to treat enriched database constraints.
- We propose the use of TAS, a methodology to develop Automated Deduction Method, to get an efficient treatment of database dependencies with a deduction method style. We intend to incorporate the notion of FD-Tree, a new representation of a set on Functional Dependencies, to the TAS methods.
- These works will be integrated around the FD^3 methodology which allows the design of relational databases in a cooperative environment. This new methodology are based in a new data model strongly based on the notion of Funcional Dependencies.

The aim of the second sub-project is to obtain sound theoretical results directed to the definition of a complete unique model for fuzzy relational deductive databases able to collect all the necessary elements. It is also important the development of the tools that permit to implement and take advantage of the mentioned model with all its capacities. The final point is to validate and prove the whole system using a big amount of data both precise and imprecise as well as a great variety of types of rule. To carry out this last task it is very important the collaboration of an expert in the field, not only for the rules definition but also for the objective criticism of the obtained results.

More specifically, this general objective include:

- To define the concept of generalized deductive rule with certainty degree. In the current state, the fuzzy relational model supports the definition of non-fuzzy rules in the sense that the rule is not affected by a degree.
- To define a deductive relational syntax built specifically for the definition of fuzzy rules.
- To study the different meanings for the accomplishment degree in a rule.

- To investigate algorithms for the optimization of rules application making more efficient all data handling, specially the queries.
- To study the inclusion of integrity rules inside the model. Integrity rules will be used to avoid inconsistency in the stored data and to hold important constraints among them without the used of external applications.
- To carry out the creation of a new type of hybrid relation called intensional-extensional in which part of the content is given in an explicit way and part of it is given by means of a set of imprecise rules.

Research tasks

In this section we describe the planed task in order to cover the above objectives:

1. *Formal study of associations*:

   The project expects to extend the initial results of this project by the sub-group of the University of Málaga to other types of dependencies, like the multivalued dependencies and, outstandingly, to fuzzy dependencies to relate it with the work made with the coordinated sub-group belonging to the investigation group of IdBIS (Intelligent Database and Information Systems) of the University of Granada.

   This generalization requires a strong work of formalization in which we are obtaining good results and that we try to go deep. For it we have worked in the design of a formal frame within the theory of coalgebras. This theory has consolidated like the ideal framework for the formalization in a great variety of fields of Computer Science (Rutten, 2000). Like examples in which coalgebraic formalizations have given rise to important advances we can emphasize the Kripke's structures, the labelled transition systems, diverse types of automata, in particular the non-determinist ones (Rutten, 2000), the programs and classes in oriented to objects languages (Reichel, 1995; Jacobs and Rutten, 1997). We try, in summary, to give a coalgebraic formalization for the different types of dependencies that allow us to unify the different theories.

   In addition in this study is included not only the dependency notion but even more flexible concepts like the association, that recently is seeing grow its interest in the scope of the data bases. So we will have a formal and uniform frame to deal with several types dependencies jointly and we will be able to take advantage of the results for the development of a new frame of logical models for data bases.

2. *Methods for manipulation of dependencies.*

   The actual works have allowed the design of algorithms that manipulate efficiently a non structured set of dependencies. The formal framework has eases the introduction of a new logic to specify and manage Functional Dependencies. The main novelty of our logic is the introduction of a new rule, the substitution rule, which facilitates the design of automated deduction methods to manage Functional Dependencies.

   These deduction methods are the basis to develop a set of executable modules that transform set of Functional Dependencies. These transformations have been integrated into some Information System intelligent tools successfully.

In a fist stage we have developed algorithms that reduce efficiently the dependencies using only the substitution rule, the new inference rule developed by the research group of this project.

In our work with Functional Dependencies we have in mind a central goal: to provide a accurate formalization and to facilitate the design of new methods and tools that manipulate database constraints in an efficient way. More specifically, we have traversed the following way:

- First we have designed a preprocessing transformation that establishes an efficient preprocessing pruning based mainly on the substitution rules, and allows for the application of any other depuration method over the new Functional Dependencies set, which is shorter than the original one.
- We have illustrated the difficulties to face the implication problem with a method directly based on logics of Functional Dependencies. We have introduced the notion of atomic-minimality, which guides the treatment of sets of Functional Dependencies in a rewriting logic. $\mathbf{SL}_{FD}$ axiomatic system is easily translated to rewriting logic and Maude 2.
- Recently, we have developed a novel method to solve the attribute closure problem. We have designed an algorithm which is based directly on the $\mathbf{SL}_{FD}$ inference system, having an automated deduction style. We have implemented the algorithm and the most efficient classical algorithm, obtaining very promising results. The complexity in the worst case is the same as the classical linear closure algorithms.
- Finally, we show how the formal techniques are used in heterogeneous and cooperative environments. We develop a tool to integrate relational schemes in an automatic way.

3. *Tools*: In the objectives we have mentioned the integration of the techniques concerning the manipulation of dependencies into a new architecture to design databases in a cooperative way. Our intention is to provide a set of intelligent tools with deduction capabilities which allows the user to manipulate a great number os information in a simple manner. Thus, we are developing a set of CASE tools that covers all the stages involved with all the database design process and introduces the deduction methods as the core of the information treatment step. The deduction methods allows a formal and efficient manipulation of the data structure at the same time that reduce the problem removing the redundancy that different user may introduce when all of them specify the same system separately.

   Besides that, these tools may be used in heterogeneous environment and allows the integration of several data sources in a unique system. To do that, the tools are extended with reverse engineering capabilities to capture the dependencies specified by the user with the notion of key. The information are stored in a Functional Dependencies data dictionary (FDD^3) which may be enriched by the database administrator in the integration process. The FDD^3 is the center element of this methodology. It may be consider as an non-structured data source which usually stores a great amount of very simple elements (the dependencies). Thus, the key of this methodology is the deduction methods which eases the task of manipulate this great amount of simple information and give an intelligent analysis of them.

Moreover we have developed a new data model named Functional Dependencies Data Model which states the specification and manipulation of information using Functional Dependencies. This data model includes a high level representation of the information using the Attribute/Dependence Diagram and a method to transform a Functional Dependencies specification into a relational database.

More concretely, we have developed two case tools which covers separately the two problems cited above.

- We have developed a CASE tool named A/D CASE (Attribute/Dependence CASE) that illustrates the practical benefits of the $FD3$ architecture. We propose the use of logics based on the notion of Functional Dependencies (FD) to allows formal specification of the objects of a data model and to conceive future automated treatment.

  The existence of a FD logic provides a formal language suitable to carry out integration tasks and eases the design of an automatic integration process based in the axiomatic system of the FD logic.

  To communicate the information collected in the global model, we introduce a *High Level Functional Dependencies* (HLFD) Data Model which is used in a similar way as the Entity/Relationship Model. The HLFD data model can be deduced automatically from a logic specification. Furthermore, it is possible to translate automatically the HLFD data model into a relational database.

- We have also used the logic algorithms to develop a new tool which integrates several ORACLE database schemes. When the user connects with the tool he introduces the identifier of the user account must be integrated and the tool, using the reverse engineering task, deduce the Functional Dependencies included in the different schemes generating a FD^3. The remove redundancy step allows the elimination of replicate and superfluous information, generating a unique and global view of the union of schemes.

  To protect the applications from this changes, a set of views are generated. These views play the role of the original tables and redirect the old queries to the new global scheme and the new tables.

  Moreover, the tool carries out an integration of the schemes and an integration of the data itself, providing a new database with a common structure and containing all the information provided in the original subschemes.

4. *Formal Techniques for Software Engineering*:

   The general framework of this project is the use of formal methods in software development. In this framework, the formalization of the static system models (data or class models) by means of dependencies it is essential. Moreover, we also consider very important to approach the formalization of the dynamic (behavior) models to obtain a complete system specification.

   In this sense, we have consider adequate to provide formalization mechanisms that can be (optionally) hidden to the software developer. In our approach, the developer may maintain the use of standard behavior models. In particular, we have considered that UML state machines are the basic behavior model. As it is well-known, UML (Unified Modeling Language) is the 'de facto' standard in object oriented software modeling.

Among the activities we have carried out in this project we have developed a formalization of the UML state machines using interval temporal logic. This approach, as far as we know, have nor been done before.

5. On the other hand, the group of Granada University has the following important tasks to carry out:

- To decide which is the deductive fuzzy model that better fits with our problems. In this sense we have studied all fuzzy operators and fuzzy implications in the literature. Once the previous choice is made, the next task is to extend the classical syntax for rules to the fuzzy case, adding an accomplishment degree. This part is already completed, but an important point now is to determine the different meanings that both the rule and the degree can have depending on the type: integrity rule, deductive rule, definition rule, dependency rule ... and to study their properties, the inclusion inside the theoretical model, and the way to store and encode them into the system.

- An important part of the work is being devoted also to the implementation of these new elements in the system, basically extending the catalog and creating new modules to create and store the rules defined. The language used to do this work is FSQL (Fuzzy SQL) developed by members of the research group IDbIS. The extension obtained for the inclusion of fuzzy rules is called DFSQL.

- Once the rules can be inserted in the system, we have developed the procedures that permit to use the rule to fill in a table, that is to compute the result of the application of a rule against the existing tables. The most interesting result of this part of the project is the possibility to define recursive rules which is a new feature inside the relational world.

Gantt chart

Tasks	1st year	2nd year	3rd year
Task 1: Theoretical Foundations of Dependencies			
1.1 To formalize the concept of minimal closure			
1.2 To extend schema. To formalize keys and antikeys			
1.3 Formal definition of normal forms			
Task 2: To extend the algebraic study to:			
2.1 Multivalued			
2.2 Fuzzy			
2.3 Rules			
Task 3: To study new order structures			
Task 4: Automatic Reasoning over dependencies			
Task 5: Development of new methodology for the design of DBs			
5.1 Implementation of diagrams of dependencies			
5.2 Tools. Integration and Purification			
5.3 Construction of the data base			
5.4 Construction of a new Model of High Level			

Tasks	1st year	2nd year	3rd year
Task 1. To define a new concept of generalized deductive rule with certainty degree.			
Task 2: To define a relational deductive syntax.			
Task 3: To study the different meanings of certainty degrees in rules			
Task 4: To investigate the algorithms for optimization of rules execution			
Task 5: To study the inclusion of integrity rules			
Task 6: Creation of a new type of intensional-extensional relation			
Task 7: To develop the algorithms for the analysis of functional dependencies			

2 Level of success achieved to date

Most of the proposed tasks are being fulfilled as expected. In the rest of the section we will describe the major achievements obtained in each of the tasks.

The publication [6] is specifically oriented to task 1. In this article, we introduce the novel algebraic formalization for the concept of schema that we propose. This formalization has the aim to be the framework to generalize the functional dependency concept and is based on the concepts of non-deterministic operators and ideals in generalizations of the lattice estructure.

Task 2 and 3 are covered in [4, 9, 5, 2]. In these papers we study in deep the concepts in which our theoretical formalization is based. The underlying idea is to provide a more general context in which several kinds of dependencies can be treated in the same way. These generalizations are finally given in terms of coalgebraic theory.

The number of articles around Task 3 is higher perhaps because it is the more close to the applications one. Concretely, in [18] we show the advantages of the novel $\mathbf{SL}_{FD}$ logic presented in [3] for the treatment of Dependencies.

In [13, 14, 17] we introduce a pre-processing transformation which removes redundancy of a given set of Functional Dependencies and allows a more efficient treatment by other well known indirect algorithms. We study in the framework of lattice theory the minimal generator of a set of functional dependencies (it mean a colateral contribution to Task 1) and we carry out an empirical study to prove the practical benefits of our approach. We develop in Prolog a automated deduction method to prune a set of Dependencies.

In [1] and in [12]we use rewriting systems to transform automatically dependencies. These articles have also colateral contributions to Task 1. In they, we consider a complexity criterion which allows us to introduce a new minimality property for sets of Functional Dependencies named *atomic-minimality.* The $\mathbf{SL}_{FD}$ logic has allowed us to develop the heart of the work, which is the use of Rewriting Logic and Maude 2 as a logical framework to search for atomic-minimality.

In [19] we study the closure operators in automatic reasoning. In [10] and [11] we present a new method to compute the closure operator for a set of attributes. The main novelty of this method is the fact that it is directly based on a logic inference system. Our method has all the advantages of using logic (solid base, flexible deduction methods, metatheory development, etc) and is the base of a novel algorithm having linear complexity in the worst case. Finally, we carry out an empirical study of the execution of our algorithm and classical closure algorithms. This study shows the best behavior of our method.

Furthermore, we have carried out an empirical study about the execution time of the Diederich and Milton closure algorithm and our closure algorithm. $\mathbf{SL}_{FD}$ Closure has had a significatively better behavior than Linear Closure. In 594 cases from 600, that is in the 99 per cent of the cases, $\mathbf{SL}_{FD}$ Closure has calculated the closure spending less time than Linear Closure. The average of time using $\mathbf{SL}_{FD}$ Closure is 110,86 and using Linear Closure is 942,05.

In [20], we present LNint-e, a novel modal temporal logic with topological semantics. LNint-e has an expressive power that is unusual in modal temporal logics, since it allows point and interval expressions, as well as the absolute and relative approches to time. This expressive power has allowed an exhaustive formalization of both the different kinds of operations of the state machines and all the transition types. LNint-e has also allowed the formalization of concurrent and non-concurrent composite states.

In [16] we have presented the logic Lni1, a first order extension of LNint-e logic. With Lni1 we could complete the formalization of our prevoius work, allowing the use of parameters in operations and events. Parameters are obviously an essential element in software modeling and has not been considered in LNint-e since this is a propositional logic.

In [8] we present results concerning the normalization of LNint-e expressions. As it is well known, normalization is a previous step in the development of deduction methods. We must point out that the definition of our normal form includes transformations that remove binary connective nestings. In this way we considerably reduce the formula management complexity.

Finally, we have began to work in Task 5 with [15] in which we develop a CASE tool that allows the integration of local subschemes designed using dependencies and in [7] we have also used the $\mathbf{SL}_{FD}$ algorithms to develop a new tool which integrates several ORACLE database schemes. It carries out an integration of the schemes and an integration of the data itself, providing a new database with a common structure and containing all the information provided in the original subschemes. The tool propose an uniform web-interface for information integration and manipulation. This tool obtains from local sub-schemes a integrated scheme using the $\mathbf{SL}_{FD}$. We use the $\mathbf{SL}_{FD}$ language as canonic model and $\mathbf{SL}_{FD}$ inference system as integration and simplification method. Moreover, this tool allows apply direct and inverse engineering techniques with the relational model.

With respect to the Granada sub-group the main results obtained can be summarized as follow:

We have organized a special session about features related to the project inside the International Congress of Information Processing and Management of Uncertainty, celebrated in Italy in July of 2004. We have already completed the algorithm used to trigger a rule and insert the results into a table. We have solved one of the possible cases to be treated in relation to the problem of rules definition in a fuzzy relational database. The algorithm calculates an intensional table (defined by means of a rule) from the fuzzy data stored in the DB [23]. We have also studied the same problem from the object-oriented perspective as can be seen in the paper [21]. The first formal results together with other results obtained previously by the group have been included in the paper [24] where real data about real state have been used to prove our system. Other results obtained have been published in the paper [22].

3 Progress indicators

In this moment two PhD students associated to the first sub-project are actively participating in different research tasks. This collaboration has produced a workshop contribution and several draft papers submitted to technical journals. We estimate that this collaboration is the beginning of theirs PhD Thesis.

The progress of the project has fruitful results that we have published in several communications.

Journal Publications

- Journals in Science Citation Index: Annals of Mathematics and Artificial Intelligence [4], Kybernetika [9], Discrete Mathematics [5]
- Other journals: Software and Systems Modeling [20][1] , Revista Iberoamericana de Inteligencia Artificial [8]

International Conferences.

- Conferences with proceedings published in Lecture Notes series: IBERAMIA: Ibero-American Conference on Artificial Intelligence [16], SBIA: XVII Brazilian Sympo-

[1] This journal edited by Springer Verlag since 2001 and it focuses on theoretical and practical issues pertaining to the development and application of software and system modeling languages and techniques. Its editorial board includes well-known researchers in the area sucha as G. Booch or D. Harel.

sium on Artificial Intelligence [1], Conferencia de la Asociación Española para la Inteligencia Artificial CAEPIA [13]

- Other conferences: Conference of Computational Methods in Sciences and Engineering - ICCMSE [14], 6th International Conference on Enterprise Information System - ICEIS [15], International Conference on Computational and Mathematical Methods in Science and Engineering - CMMSE [17],

International Workshops

Theory and Applications of Relational Structures as Knowledge Instruments. Meeting of the COST Action 274 [18]

National Conferences

Primer Congreso Conjunto de Matemáticas RSME-SCM-SEIO-SEMA [12], XV Congreso Nacional Usuarios de Oracle QUORE [7],

The main researcher of the sub-group of Granada is tutoring a PhD Thesis with the aim to study and develop all the types of rules presented in the project. The PhD student is a researcher of the Instituto Andaluz de Astrofisica (ESA) and the research project he is included in is Exploration of the Solar System.

References

[1] Gabriel Aguilera, Pablo Cordero, Manuel Enciso, Angel Mora, and Inmaculada P. de Guzmán. A non-explosive treatment of Functional dependencies using rewriting logic. *Lecture Notes in Artificial Intelligence, Springer-Verlag*, 3171, 2004, pp. 31–40.

[2] Pablo Cordero, Inmaculada de las Peñas, Gloria Gutiérrez and Javier Martínez. Non determinism, coalgebras and multisemilattices. Submited to *Theoretical Computer Science*.

[3] Pablo Cordero, Manuel Enciso, Inmaculada P. de Guzmán, and Angel Mora. SLFD logic: Elimination of data redundancy in knowledge representation. *Lecture Notes in Artificial Intelligence 2527, Springer-Verlag*, pages 141–150, 2002.

[4] Pablo Cordero, Gloria Gutiérrez, Javier Martínez, and Inmaculada P. de Guzmán. A new algebraic tool for automatic theorem provers. *Annals of Mathematics and Artificial Intelligence,* 42(4):369-398, 2004.

[5] Pablo Cordero, Javier Martínez, Gloria Gutiérrez, and Inmaculada P. de Guzmán. Generalizations of lattices looking at computation. *Discrete Mathematics,* 295(1-3):107-141, 2005.

[6] Pablo Cordero, Angel Mora, Inmaculada P. de Guzmán and Manuel Enciso. Non deterministic ideal operators: An adequate tool for formalization in Data Bases. *Discrete Applied Mathematics*, in review.

[7] Manuel Enciso, Angel Mora, J. Marco, José Millán. Herramienta web para integrar la información de diferentes esquemas Oracle. *XV Congreso Nacional Usuarios de Oracle. Octubre de 2005.*

[8] Manuel Enciso, Inmaculada P. de Guzmán, and Carlos Rossi. Una forma normal temporal independiente del método de deducción. *Inteligencia Artificial,* 8, 23 (2004) pp. 27–46.

[9] J. Martínez, P. Cordero, G. Gutiérrez, and Inmaculada P. de Guzmán. Restricted ideals and the groupability property: tools for temporal reasoning. *Kybernetika*, 39(5):521-546, 2003.

[10] Angel Mora, Gabriel Aguilera, Manuel Enciso, Pablo Cordero, and Inmaculada P. de Guzmán. Closure via Functional Dependence Simplification. *Sometido a Journal Acta Informatica. Elsevier. JUNE 2005*,

[11] Angel Mora, Gabriel Aguilera, Manuel Enciso, Pablo Cordero, and Inmaculada P. de Guzmán. A new closure algorithm based in logic: SLfd-Closure versus classical closures. *Sometido a Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2005). Lecture Notes in Computer Science. JUNE 2005*,

[12] Angel Mora, Gabriel Aguilera, Manuel Enciso, Pablo Cordero, and Inmaculada P. de Guzmán. SLfd-logic to obtain atomic-minimality in Functional Dependencies. *Sesión especial de Herramientas matemáticas para la computación. Primer Congreso Conjunto de Matemáticas MAT.ES (RSME-SCM-SEIO-SEMA). Valencia 2005.*

[13] Angel Mora, Manuel Enciso, Pablo Cordero, and Inmaculada P. de Guzmán. An efficient preprocessing transformation for functional dependencies sets based on the substitution paradigm. *Lecture Notes in Artificial Intelligence, Springer-Verlag*, 3040, 2004, pp. 136–146.

[14] Angel Mora, Manuel Enciso, Pablo Cordero, and Inmaculada P. de Guzmán. Minimal generator of a non-deterministic operator.. *International Conference of Computational Methods in Sciences and Engineering - ICCMSE 2004. Lecture Series on Computer and Computational Sciences. VSP international Science Publishers. The Netherlands. Series ISSN 1573-4196 ISBN: 90-6764-418-8.* , Volume 1, pp. 396-399.

[15] Angel Mora, Manuel Enciso, Pablo Cordero, Inmaculada P. de Guzmán and José Guerrero. A/D CASE: A new heart for FD3. *6th International Conference on Enterprise Information System. ICEIS 2004 ICEIS PRESS. Proceedings of the 6th International Conference on Enterprise Information System. ISBN: 972-8865-00-7 PORTO - PORTUGAL, 2004.*

[16] Manuel Enciso, Carlos Rossi, Angel Mora. A first order temporal logic for behavior representation IBERAMIA *Ibero-American Conference on Artificial Intelligence. Lecture Notes in Artificial intelligence, vol 3315 pp 408–418 Mexico, 2004.*

[17] Angel Mora, Manuel Enciso, Pablo Cordero, and Inmaculada P. de Guzmán. SLfd logic in Prolog to remove redundancy in database. *International Conference on Computational and Mathematical Methods in Science and Engineering. V - Proceedings of the CMMSE 2005. ISBN: 84-609-4844-7*, pp. 390–395.

[18] Angel Mora, Manuel Enciso, Pablo Cordero, and Inmaculada P. de Guzmán. Logic treatment of Functional Dependencies in the Relational Model, yet another novelty. *Theory and Applications of Relational Structures as Knowledge Instruments. Meeting of the COST Action 274 Málaga. March 31 - April 2, 2005.*

[19] Francisco Rodríguez, Pablo Cordero, and Inmaculada P. de Guzmán. Operadores de cierre y razonamiento automático. *In Actas del III Encuentro Andaluz de Matemática Discreta 2003,* pp. 83–86.

[20] Carlos Rossi, Manuel Enciso, and Inmaculada P. de Guzmán. Formalization of UML state machines using temporal logic. *Software and Systems Modeling,* 3, 1 (2004) 31-54.

[21] F. Berzal, N.Marin, O.Pons, M.A. Vila A framework to build fuzzy object-oriented capabilities over an existing database using classical object-oriented concepts *Advances in fuzzy object oriented databases: modelling and applications*, chapter 7 . 2004

[22] F. Berzal, N.Marin, O.Pons, M.A. Vila Managing fuzziness on conventional object oriented platforms *Advances in fuzzy database technology. Int. Journ. of Intelligent Systems.* 2005

[23] I.Blanco, M.J. Martin-Bautista, O. Pons, M.A. Vila A Tuple-Oriented Algorithm for Deduction in a Fuzzy Relational Database *Int.Journ. of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 11. 2003

[24] C.D. Barranco, J. Campaña, J.M. Medina, O. Pons InmoSoftWeb: A web based fuzzy application for real state *Lecture Notes in Computer Science*, vol. 3034. 2004