

## UN ENFOQUE PRAGMÁTICO PARA LA REPRESENTACIÓN DEL CONOCIMIENTO BASADO EN INCERTIDUMBRE: TÉCNICAS DE BASES DE DATOS Y ESCENARIOS DE APLICACIÓN

**Autores:** Miguel Ángel Sicilia Urbán  
Laboratorio DEI, Departamento de Informática, Universidad Carlos III Leganés (Madrid)  
msicilia@inf.uc3m.es  
Intelligent Software Components (iSOOCO) Madrid  
masicilia@isoco.com

Elena García Barriocanal  
Dpto. CC. Computación, Universidad de Alcalá  
Escuela Politécnica. Campus Universitario. 28871, Alcalá de Henares Madrid  
elena.garciab@uah.es

José R. Hilera González  
Dpto. CC. Computación, Universidad de Alcalá Escuela Politécnica.  
Campus Universitario. 28871, Alcalá de Henares (Madrid)  
Jose.hilera@uah.es

Roberto Barchino Plata  
Dpto. CC. Computación, Universidad de Alcalá Escuela Politécnica.  
Campus Universitario. 28871, Alcalá de Henares (Madrid)  
Roberto.barchino@uah.es

**Resumen:** Para llegar a ser un reflejo adecuado de la capacidad cognitiva humana, las técnicas de representación del conocimiento debe tener en cuenta diferentes aspectos relacionados con la noción de incertidumbre. En concreto, podemos tomar elementos de bases de datos borrosas y consulta flexibles y combinarlos con técnicas estándar de desarrollo, para conseguir un marco de trabajo útil y práctico de propósito general. En este trabajo describimos una versión preliminar de nuestro marco de trabajo, que combina modelado conceptual mediante extensiones de UML con especificaciones de más bajo nivel en cálculo relacional, que son directamente trasladables a bases de datos convencionales. Describimos además dos casos de estudio para ilustrar la aplicabilidad de nuestras técnicas.

**Palabras clave:** Representación del Conocimiento; Conocimiento Basado en Incertidumbre; Bases de Datos Borrosas

**Abstract:** In order to become an adequate reflection of the human cognitive ability, knowledge representation techniques should take into account different aspects related with the notion of uncertainty. Specifically, we can take elements from fuzzy databases and flexible queries and combine them with standard development techniques to come up with a useful and pragmatic general-purpose framework. In this work, we describe the preliminary version of our framework, which combines conceptual modeling through UML extensions with lower-level relational calculus specifications, which are translatable to conventional database schemas in a straightforward way. We describe also two case studies to illustrate the applicability of our approach.

**Keywords:** Knowledge Representation; Uncertainty-Based Knowledge; Fuzzy Databases

## 1. Introducción: Representaciones del Conocimiento e Incertidumbre

Las representaciones del conocimiento pueden considerarse como *sustitutos* (Davis, 1993) de sistemas u objetos externos tangibles o intangibles orientados a algún tipo de proceso de razonamiento interno en un agente (humano o artificial). Aunque una fidelidad exacta entre los mundos interno y externo es inalcanzable, las representaciones del conocimiento deben tener en cuenta características fundamentales relacionadas con la incertidumbre, como la *vaguedad* y la *imprecisión*, que son inherentes a los procesos cognitivos humanos; véase como ejemplo (Rocha, 1999). Algunos de los sistemas clásicos de representación del conocimiento (por ejemplo, lógica formal, ontologías o sistemas de marcos) tienen ciertas dificultades a la hora de modelar este tipo de características del conocimiento, y como reacción, han comenzado a aparecer sistemas de inteligencia artificial que admiten diferentes tipos de conocimiento impreciso, tales como los sistemas expertos borrosos, que pueden construirse con herramientas específicas del tipo de MILORD (Sierra, 1992) o FUZZYJESS<sup>1</sup>. De entre los diferentes modelos matemáticos de la información basada en incertidumbre (Klir, 1999), nosotros nos centramos en la teoría de conjuntos borrosos y sus aplicaciones al campo de las bases de datos. Por tanto, y según la clasificación que puede encontrarse en (Smets, 1997), nos limitamos a tratar conocimiento vago, impreciso o aproximado (pero no erróneo o inconsistente).

El resto de este artículo se estructura de la siguiente manera: la Sección 2 describe nuestro enfoque para el desarrollo de aplicaciones que incorporan una representación de conocimientos imprecisos almacenados en bases de datos relacionales; la Sección 3 describe dos casos de estudio en dominios bien diferentes y, finalmente, en la Sección 4 presentamos las conclusiones fundamentales y esbozamos nuestras líneas de trabajo futuras.

## 2. Objetivos Pragmáticos de la Representación del Conocimiento Basadas en la Incertidumbre

Nuestra propuesta consiste en un sistema de representación del conocimiento compuesto por dos capas (en el sentido de técnicas de modelado del conocimiento); una para el modelado conceptual y otra de representación a más bajo nivel, para la cual utilizamos una notación semi-formal de cálculo relacional. Una vez definida la representación mediante cálculo relacional, la implementación en bases de datos es un paso directo. La ventaja de utilizar la notación de cálculo es que, por su naturaleza declarativa, nos permite razonar sobre relaciones y consultas de manera precisa y próxima al diseño lógico de la base de datos, evitando la descripción detallada prematura de esquemas relacionales mediante diagramas o tablas.

Aunque el objetivo último de nuestro sistema de representación de conocimiento basado en incertidumbre es constituir una base para el paradigma de Zadeh de la "Computación con palabras" ("Computing with

---

<sup>1</sup> Disponible en el FUZZYJ TOOLKIT, <[http://ai.iit.nrc.ca/IR\\_public/fuzzy](http://ai.iit.nrc.ca/IR_public/fuzzy)>

Words”), lo conseguido hasta el momento y nuestros objetivos a medio plazo se limitan a conseguir un marco de trabajo (*framework*) que permita construir aplicaciones eficientes basadas en información de naturaleza imprecisa. En concreto, nuestra propuesta de representación del conocimiento trata de manejar la riqueza de la semántica del lenguaje natural en lo que concierne a los niveles de *vaguedad* e *información incompleta*, definiendo para ello un marco adecuado de *metadatos*. Desde nuestro punto de vista, el punto central de la representación del conocimiento es la construcción de *sustitutos* para algunas estructuras semánticas del lenguaje natural, además de los habituales *sustitutos* de las entidades del mundo real (semánticas referenciales). Aunque se está todavía lejos de una teoría general de semántica obtenida a partir del análisis empírico del acto de comunicación humana – como la iniciada en (Rieger, 1979) –, debemos guiar nuestra atención a modelos restringidos dependientes del dominio que puedan ser útiles en aplicaciones del mundo real. Desde una perspectiva pragmática, las actuales técnicas sólo consiguen algunas formas de consulta flexibles que manejan términos lingüísticos predefinidos o categorías, pero que sin embargo, son lo suficientemente potentes para permitir algunas formas poco profundas de predicción y razonamiento. El resto de esta Sección describe el estado actual de nuestro marco de trabajo en la dirección apuntada.

## 2.1. Modelado Conceptual, Extensiones de UML y Cálculo Relacional

A la hora de especificar la representación del conocimiento en un nivel conceptual, el énfasis debe ponerse en la facilidad de comprensión por parte de conocedores del dominio. Para ello, hemos definido nuestra propia versión extendida del lenguaje de modelado UML (OMG, 1999) como herramienta de modelado conceptual, para la representación de las bases de datos relaciones subyacentes y las extensiones de las interfaces de programación de aplicaciones (*application programming interfaces, APIs*) comunes de bases de datos. De este modo, buscamos construir nuestra propia representación del conocimiento generalizada lo más cercana posible a las actuales tendencias comerciales de ingeniería del software. La Figura 1 muestra un ejemplo de modelado extendido para una aplicación Web.

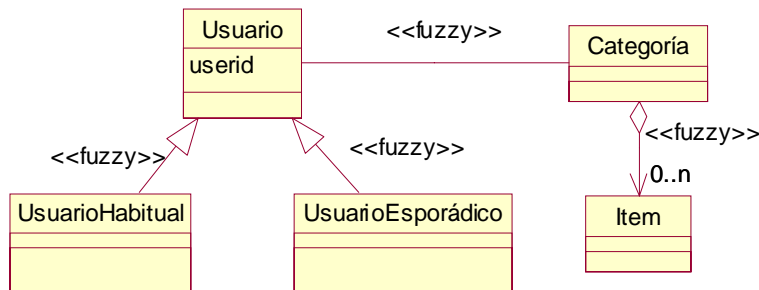


Figura 1. Ejemplo de Modelo UML Extendido para Modelado de Imprecisión

En la Figura 1 aparecen ejemplos de dos tipos de relaciones extendidas del metamodelo del estándar UML. Hemos utilizado el mecanismo de extensión de los estereotipos de UML, definiendo `<<fuzzy>>` como subclasificador de relaciones. Por un lado, las asociaciones o agregaciones borrosas representan

relaciones borrosas entre conjuntos borrosos o nítidos. Por ejemplo, un usuario puede estar interesado en varias categorías de libros con diferentes “grados” de interés en cada una. De igual modo, los libros contenidos en cada categoría pueden estarlo con diferentes gradaciones. Por otro lado, y basándonos en la semántica de conjunto que implícitamente define una clase respecto a sus instancias, hemos especializado la relación de generalización para indicar que las instancias de una clase padre pueden ser instancias de clases derivadas de manera borrosa con un cierto grado. Por ejemplo, un usuario puede clasificarse como habitual en un grado alto, digamos 0.9 en una escala [0..1].

Una vez definidos los modelos conceptuales, pasamos a la descripción detallada del almacenamiento mediante cálculo relacional. Esas descripciones contendrán los elementos típicos de las bases de datos relacionales, pero orientados a la representación de diferentes tipos de información imprecisa. Por el momento, nos hemos ceñido a los “paradigmas” más habituales (Chen, 1998), concretamente a los dos siguientes:

- Marco de trabajo basado en relaciones borrosas (a “nivel de tupla”). Se permite a una tupla la pertenencia a una determinada relación con un grado [0,1], en lugar del habitual {0,1} (“está o no está”).
- Marco de trabajo “Basado en Similitud”. Cada dominio de la base de datos puede asociarse a una relación de similitud, y cada valor de atributo puede guardar un subconjunto del dominio (en lugar de un solo elemento). Esto es, dado un dominio finito  $D_i$ , podemos definir un atributo en su conjunto potencia  $2^{D_i}$ . La relación de similitud permite comparar valores y no es más que una correspondencia  $D_i \times D_i \rightarrow [0,1]$  con reflexividad, simetría y transitividad.

En el primer caso, los esquemas deben representarse mediante tablas de metadatos, en concreto una relación  $RB = \{t \mid t_{relation}, t_{\mu-field}\}$  que marca los campos que almacenan los valores de pertenencia a las relaciones borrosas. Siguiendo un ejemplo anterior, la tupla (*esporádicos, grado*) indica que el atributo grado guarda la pertenencia de los usuarios al subconjunto de los esporádicos. En el segundo caso, tendremos que definir una relación con la definición de los atributos del tipo extendido de la forma  $P = \{t \mid t_{name} \in String \wedge t_{type} \in types \wedge t_{multi} \in \{true, false\}\}$ , donde *types* representa un conjunto de tipos que incluye tipos convencionales (por ejemplo, los del lenguaje Java, incluyendo el tipo cadena *String*), *name* es el nombre de un atributo y el atributo *multi* define si el atributo es multivaluado o no. Así, un atributo basado en similitud se definiría mediante una tupla (*fidelidad, type-s1, false*), suponiendo que estamos describiendo la fidelidad de un cliente en un *sitio* de comercio electrónico. El tipo *type-s1* tendrá una relación de similitud (o proximidad, relajando la relación transitiva) asociada, almacenada en tablas adicionales. Con las definiciones anteriores, podemos expresar los elementos del conjunto borroso en la tabla “habituales”, cuyo valor para  $\mu$ -field en RB es  $g$  como  $\{t \mid t \in habituales \wedge t_g > 0\}$ . También podemos obtener los usuarios más fieles que el usuario *u1* mediante la expresión  $\{t \mid t \in usuarios \wedge t_{fidelidad} > f1\}$ , dada *f1* la fidelidad para *u1*, y suponiendo transparente la ejecución de la desigualdad a partir de la relación de similitud almacenada.

## 2.2. Extensión de Mecanismos de Base de Datos

Puesto que existen investigaciones maduras en cuanto a modelos de datos/bases de datos borrosas de nivel físico, lógico y conceptual (véase, por ejemplo, (Medina, 1994) o (Yacizi, 1999)), estas pueden extenderse para habilitar la representación de cierto *conocimiento orientado al proceso* adicional, dando como resultado una representación del conocimiento generalizada, que es capaz de modelar un amplio espectro de conocimiento basado en incertidumbre utilizando atributos basados en similitud o proximidad, relaciones borrosas o distribuciones de posibilidad. De este modo, nos ajustamos a la idea genérica de una representación del conocimiento como *módulo ask/tell* (definición operativa) con la siguiente semántica:

- TELL(K,h). Dada una base de conocimientos K, se añade el hecho dando como resultado una base de conocimientos K' aumentada.
- Ask(K, h). Consultamos a la base de conocimiento K sobre el hecho h. En nuestro caso, las respuestas siempre se expresan con vaguedad.

La Figura 2 muestra la arquitectura general de capas de nuestra aproximación.

La semántica de TELL se implementa mediante la adición directa de conocimiento a través de nuestras propias interfaces extendidas del API de acceso a datos JDBC. Estas nuevas interfaces se han aumentado de manera que su uso sea lo más natural posible para programadores de base de datos. Respecto a la semántica Ask, modelamos el conocimiento mediante reglas de aplicación (que pueden entenderse como reglas de negocio – *business rules* – en la mayoría de las aplicaciones). Estas reglas pueden implementarse de dos maneras diferentes:

- Escribiéndolas de manera explícita en la lógica del código de la aplicación (mediante el uso de consultas flexibles, con o sin cuantificación).
- O bien empotrando un motor de encadenamiento de reglas al estilo de un sistema experto clásico, como puede ser FUZZYJESS.

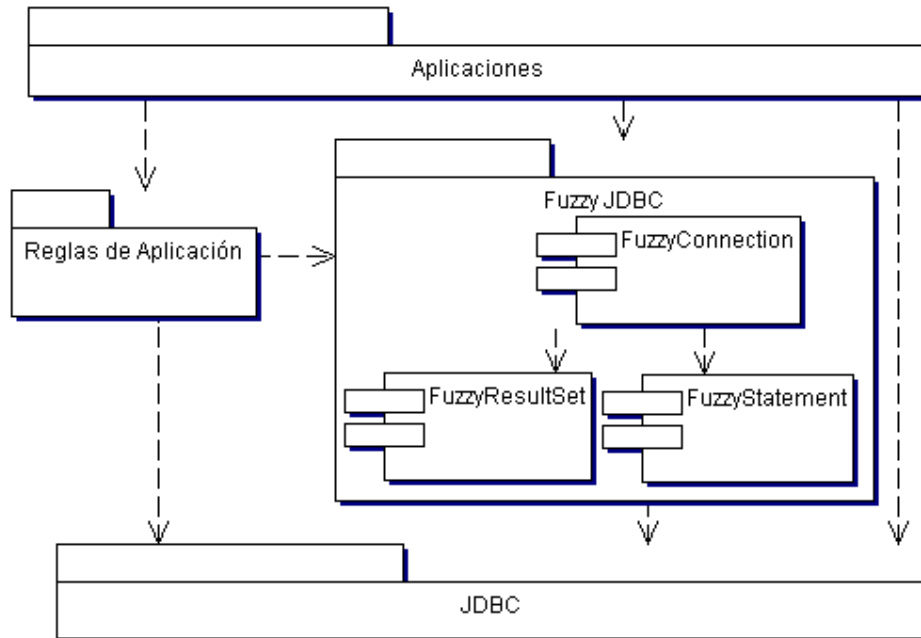


Figura 2. Un Marco de Trabajo para Aplicaciones Basadas en Conocimiento Impreciso.

Las reglas adoptan la forma de reglas clásicas de inferencia borrosa – tal y como se describen en (Tsoukalas, 1997, p.105-144) –, o pueden consistir en la ejecución de un algoritmo que implica la iteración sobre resultados de consultas flexibles.

### 3. Escenarios de Aplicación

Hemos probado los sistemas fundamentales de lógica borrosa para modelos relacionales que acabamos de describir en dos áreas de aplicación distintas que pasamos a esbozar brevemente: personalización basada en modelos de usuario para aplicaciones Web e Historias Clínicas.

#### 3.1. Personalización de aplicaciones Web

La personalización en Web requiere construir tres modelos esenciales: el Modelo de Usuario (MU), el Modelo de Contenidos (MC) y el Modelo de Adaptación (MA). En esencia, el MA describe cómo los contenidos (fragmentos de HTML, XML combinado con XSL u otros “medios”) descritos en el MC se adaptan para personalizar la experiencia del usuario de acuerdo a su perfil, que se recoge en el MU. El paradigma comercial más habitual del MA es el de la personalización “basada en reglas”, que se consigue mediante la definición de reglas del tipo: “para los usuarios que cumplen la restricción  $f(u)$ , muestra los contenidos que cumplen la restricción  $g(c)$ ”. Debido a que la descripción de las restricciones sobre el usuario suele incluir reglas de clasificación, de la forma “si el usuario cumple la condición  $w(u)$  entonces el usuario se clasifica en la categoría  $W$ ”, suele incorporarse un motor de inferencia para resolver las reglas

encadenadas en ejecución, con una arquitectura similar a la de CAPE (Inder, 2000).

Como forma de extracción de conocimiento resumido inferido hemos utilizado consultas borrosas cuantificadas junto con técnicas borrosas de consulta flexible, tanto para disparar eventos externos como para actualizar la base de conocimiento– los resultados preliminares se describen en (Sicilia, 2000a). Dada la alta sensibilidad de las aplicaciones Web con gran número de usuarios a los accesos a base de datos, la elección de la representación de la incertidumbre está basada en el análisis sobre el rendimiento de las consultas en diferentes representaciones básicas de información borrosa, que pueden consultarse en (Sicilia, 2000b).

### 3.2. Modelos de Historia Clínica

Los modelos de Historia Clínica informatizada han cristalizado en propuestas estandarizadas de modelos de un amplio alcance (que abarcan desde la admisión de pacientes hasta el registro detallado de las analíticas). Por el momento, sólo hemos trabajado en la extensión de modelos existentes para dar cabida a ciertas formas de información imprecisa. En concreto, hemos considerado los modelos de HL7 RIM (*Reference Information Model*) y el GEHR.

El HL7 RIM proporciona un soporte limitado para representar incertidumbre mediante dos tipos de datos predefinidos: *UncertainValueNarrative* (UVN) y *UncertainValueProbabilistic* (UVP). El primero de ellos sirve para asociar al valor de un tipo de datos una etiqueta – por ejemplo “aproximadamente” o “probablemente” – que actúa a modo de *cualificador* de confianza; el problema es que las etiquetas no están normalizadas y por tanto su tratamiento automatizado se hace inviable. El segundo de ellos permite anotar los valores con una estimación de probabilidad subjetiva de los mismos. En ambos casos son medidas de creencia, pero no proporcionan soporte al almacenamiento de información lingüística de forma directa. La extensión del RIM consistiría en la inclusión de nuevos tipos en la categoría *RIM\_Datatypes*. En concreto, nuestra propuesta incluye un tipo concreto *LIST<FS>*, que se obtiene a partir del tipo genérico *LIST:Sequence* definido en RIM. Las listas se definen como secuencias ordenadas de conjuntos borrosos representados por el tipo *FuzzySet* (FS), directamente derivado del tipo abstracto *ANY* definido en RIM. El tipo FS toma como valores funciones parametrizadas típicas de representación de conjuntos borrosos (trapezoidales, triangulares, funciones-S, etc.). De igual forma, definimos un tipo *FuzzyHedge* (FH) para modelar modificadores borrosos como los típicos “muy” o “bastante”. Por último, definimos un tipo *FuzzyValue* (FV) que puede contener una etiqueta lingüística de entre las que pueden generarse a partir de una representada mediante el tipo *LIST<FS>*. Cada *FV* tiene una lista *LIST<FS>* asociada de la cual puede tomar valores, bien directamente o bien prefijándole uno o varios modificadores tomados de un conjunto *SET<FH>*. La lista y el conjunto mencionados se modelan como metadatos (*componentes* en RIM) asociados al tipo FV. El tipo FV permite representar números borrosos (Tsoukalas, 1997, p.77-79) para las diferentes manipulaciones, incluyendo las consultas con cuantificadores borrosos. A la hora de trasladar el sistema de

tipos RIM a una base de datos, los vocabularios de etiquetas lingüísticas pueden traducirse fácilmente en tablas dentro del marco de trabajo basado en relaciones borrosas, aunque en los casos de elementos *LIST<FS>* que funcionen como cualificadores u opiniones (como en el caso de la lista: excelente, bueno, regular, malo, pésimo) puede ser conveniente el almacenarlas como tales etiquetas definiendo entre ellas una relación de similitud para poder compararlas y agregarlas. Por otro lado, una versión modificada de la definición del tipo RIM *ParametricProbabilityDistribution* (PPD) permite el almacenamiento de distribuciones de posibilidad.

El modelo GEHR (*Good Electronic Health Record*) es una especificación de una arquitectura para el almacenamiento de la Historia Clínica, en la cual la información clínica se estructura en definiciones llamadas *arquetipos* que pueden definir los expertos. Los arquetipos son definiciones estructurales de conceptos clínicos en términos de restricciones sobre el uso de objetos GEHR. El conjunto de tipos de datos básicos de GEHR (incluidos en el *GEHR Object Model*, GOM) no proporciona soporte directo para conceptos borrosos y tampoco pueden modelarse de manera adecuada mediante los conceptos de *transacción* u *organizador*. Sin embargo, el lenguaje de metadatos (expresado en XML) que GEHR proporciona para la definición de arquetipos es suficientemente abierto como para incorporar dichos conceptos. Basta con ampliar el conjunto de tipos básicos de GEHR. Tomemos como ejemplo el siguiente fragmento de definición de arquetipo para presión sanguínea:

```
<complexType name="diastolicItem"
  base="gom:G1_HIERARCHICAL_VALUE" derivedBy="restriction">
  <element name="name" type="bp:diastolicName"/>
  <element name="value" type="bp:bpDataValue"/>
  <element name="reference" type="gom:G1_STATISTICAL_REFERENCE"
    minOccurs="0" maxOccurs="1"/>
</complexType>
...
<complexType name="bpDataValue" base="gom:G1_QUANTITY"
  derivedBy="restriction">
  <element name="value">
    <simpleType base="gom:G1_REAL" derivedBy="restriction">
      <minInclusive value="0"/>
      <maxInclusive value="1000"/>
    </simpleType>
  </element>
...
```

En él observamos la definición de una medida de con una referencia estadística cuyo valor se define mediante un tipo *bpDataValue* que en último término se define en función de un tipo básico de GOM, en este caso un valor real. Estos tipos se definen con la ayuda del lenguaje Eiffel, y podrían por tanto extenderse con nuevas clases para los tipos de datos, con conceptos similares a los mencionados sobre el RIM de HL7.

#### 4. Conclusiones y Trabajo Futuro

El desarrollo de aplicaciones dirigidas por una representación del conocimiento basado en la incertidumbre puede conseguirse mediante la extensión de los esquemas de bases de datos relacionales mediante técnicas



soportadas por una base teórica sólida. Los modelos conceptuales se ven enriquecidos significativamente por la inclusión de tipos capaces de representar la incertidumbre, aumentando la expresividad semántica de las representaciones del conocimiento. Las extensiones utilizadas han demostrado ser útiles tanto en el diseño de las aplicaciones como en las reuniones con usuarios potenciales, que comprenden fácilmente las herramientas de modelado con incertidumbre.

Nuestro trabajo futuro se centra en hacer más completas y flexibles las interfaces de programación actuales, incorporando técnicas adicionales de representación borrosa y/o consulta flexible. También trabajamos en ampliar las representaciones con un marco de trabajo "basado en distribuciones de posibilidad", y en la formalización de nuestras extensiones de UML.

En el horizonte de más largo plazo, nos planteamos la integración de los modelos borrosos actuales con otros modelos de incertidumbre, para proporcionar un conjunto de herramientas del cual el desarrollador pueda seleccionar las más apropiadas para su aplicación.

## Referencias

CHEN, G. 1998. *Fuzzy Logic in Data Modeling : Semantics, Constraints, and Database Design*. The Kluwer International Series on Advances in Database Systems, Kluwer Academic Pub, August 1998.

DAVIS, R., *et al.* 1993. What is knowledge representation?. *AI Magazine*, 1993, n.14, p.17-33.

INDER, R. 2000. A tool for Internet-oriented knowledge based systems. En *Proceedings of the ACM symposium on Applied computing*, 2000, vol.1, p. 34 – 39.

KLIR, G.J., WIERMAN, M.J. 1999. *Uncertainty-Based Information: Elements of Generalized Information Theory*. Springer Verlag, 2nd ed., 1999.

MEDINA, J.M., *et al.* 1994. Fuzzy Knowledge Representation in Relational Databases. 1994. Technical Report DECSAI-94112, Universidad de Granada, November.

OMG (Object Management Group). 1999 [en línea]. *OMG Unified Modeling Language Specification, Version 1.3*. June 1999. Disponible en <[http://www.omg.org/technology/documents/formal/unified\\_modeling\\_language.htm](http://www.omg.org/technology/documents/formal/unified_modeling_language.htm)>

RIEGER, B., 1979. Fuzzy Structural Semantics. On a generative model of vague natural language meaning. En Trappl, R., Hanika, P., Pichler, F.R. (eds.): *Progress in Cybernetics and Systems Research*, 1979, Vol. V, Washington/ New York/ London (Wiley & Sons), p. 495-503.

ROCHA, L.M. 1999. Evidence sets: modeling subjective categories. *International Journal of General Systems*, vol. 27, p. 457-494.

SICILIA, M.A, DODERO, J.M. 2000a. User Stereotype Modelling for Approximate Reasoning-based Adaptive Enterprise Portals. En Proceedings of the 10th European-Japanese Conference on Information Modelling and Knowledge Bases. Saariselka, Finland, 2000, p.140-147.

SICILIA, M.A. *et al.*, 2000b. Representación de conjuntos y cuantificadores lingüísticos borrosos en esquemas relacionales. En X Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF), 2000, Sevilla, p.161-166.

SIERRA, C., GODO, L. 1992. Modularity, uncertainty and reflection in MILORD II. In Proceedings of the 1992 IEEE International Conference on Systems, Man and Cybernetics, Chicago, Illinois, October 1992, vol. 1, p. 255--260.

SMETS, P. 1997. Imperfect information : Imprecision - Uncertainty. En Uncertainty Management in Information Systems. From Needs to Solutions. MOTRO, A. SMETS, P. (eds.), Kluwer Academic Publishers, 1997, p. 225-254.

TSOUKALAS, L.H., UHRIG, R.E. 1997. *Fuzzy and Neural Approaches in Engineering*, Adaptive and Learning Systems for Signal Processing, Communications and Control Series, John Wiley & Sons. 1997.

YACIZI, A., GEORGE, R. 1999. *Fuzzy Database Modeling*. Studies in Fuzziness and Soft Computing Series, Physica-Verlag, 1999.