



ELEMENTOS GRAMATICALES Y CARACTERÍSTICAS QUE DETERMINAN APLICACIONES CON REQUERIMIENTOS DIFUSOS

(GRAMMATICAL ELEMENTS
AND CHARACTERISTICS THAT
DETERMINE APPLICATIONS WITH
FUZZY REQUIREMENTS)

Resumen

En este artículo se establecen las bases para la determinación de la necesidad de uso de consultas difusas en aplicaciones informáticas. Se estudian aspectos teóricos, lingüísticos y experiencias prácticas. Se definen siete características fundamentales que deben presentar tales aplicaciones. Se evidencian siete clases de elementos gramaticales del lenguaje natural que distinguen los requerimientos de información que se deben implementar mediante consultas difusas.

Palabras claves: *SQLf, Consultas Difusas, Requerimientos Vagos, Gramática del Lenguaje Natural*

Abstract

In this work we establish the basis for determining the need of fuzzy queries in information systems. We have studied fuzzy logic theoretical aspects and natural language linguistic concepts as well as practical experiences. We have defined seven fundamental characteristics that such applications must have. Also we put in evidence seven grammatical elements that would identify information requirements to be implemented by means of fuzzy queries use.

Keywords: *SQLf, Fuzzy Querying, Vague Requirement, Natural Language Grammar*

■ Rosseline Rodríguez
email: crodrig@usb.ve

■ Leonid Tineo
email: leonid@usb.ve

Departamento de Computación
Universidad Simón Bolívar
Valle de Sartenejas, Baruta
Caracas-VENEZUELA

Fecha de Recepción: 10 de octubre de 2008

Fecha de Aceptación: 19 de enero de 2009

Introducción

Las bases de datos tradicionales sufren del problema de rigidez [7] [25] [12], en el sentido de no poder representar información imperfecta y requerimientos vagos que retornen respuestas discriminadas basados en preferencias de usuarios. En el pasado se ha demostrado que la aplicación de la teoría de conjuntos difusos [29] para la representación de términos del lenguaje natural [30] es el enfoque más adecuado para resolver el problema de rigidez [2] [7] [26]. Se han realizados diversos esfuerzos [2] [9] [11] [20] [21] [22] [23] [28] para extender los sistemas de bases de datos relacionales con el uso de los conjuntos difusos. Entre ellas, SQLf [2] [15] [16] [19] tiene la virtud de proveer la más amplia gama de estructuras de consulta que permiten el uso de condiciones difusas y de ceñirse a los estándares de SQL. Recientemente se ha publicado un libro que provee una cobertura amplia de los tópicos más importantes, conceptos, tendencias y tecnologías de bases de datos difusas [13], en éste se incluyen aportes recientes de investigación de los más importantes expertos en el área.

Desde 1998, en la Universidad Simón Bolívar [26], se ha venido trabajando para construir un manejador de base de datos que utilice la lógica difusa como herramienta para flexibilizar las consultas a bases de datos existentes. Estos esfuerzos abarcan desde la actualización de SQLf según los estándares de SQL [15] [16] [19], así como también, la implementación de un servidor para el manejo de bases de datos difusas [18] y diversas aplicaciones realizadas usando este sistema [1] [4] [5] [6] [8] [17].

La creación de Sistemas de Consultas Difusas a Bases de Datos es un área abierta de investigación [13]. Sin embargo, al igual que otros tópicos en el pasado, hay cierta resistencia hacia este tema. Una de las razones de ello es que se presupone que es muy alto el volumen de cómputo asociado al cálculo de grado de satisfacción de condiciones en Lógica Difusa [3] [22] [19]. Por otro lado, como las aplicaciones clásicas que usan bases de datos son resueltas con los sistemas de consulta basados en la Lógica Booleana [7] [25], se piensa que un Sistema de Consultas Difusas tendría el riesgo de quedarse simplemente como un instrumento interesante de laboratorio pero sin utilidad práctica.

El problema de la sobrecarga computacional del procesamiento de consultas difusas está siendo aún estudiado. Sin embargo, se tienen resultados teóricos prometedores, como lo es el Principio de Derivación

propuesto inicialmente en [3], para consultas relacionales difusas de bloque simple. Este principio ha sido posteriormente estudiado y extendido para cualquier tipo de consulta en SQLf [15] [16] [27]. Aplicando este principio, se ha desarrollado un Sistema de Consultas Difusas denominado SQLfi, (SQL flexible en internet) [18], el cual está en Oracle 9i y actualmente se está extendiendo para dar soporte a otros manejadores [10].

Por otro lado, en cuanto a las aplicaciones, la realidad actual es otra: con la expansión de los grandes sistemas masivos de información y comercio electrónico a través de la red global de informática a lo ancho del mundo (WWW), se hace de gran utilidad la disponibilidad de Sistemas de Consultas Difusas a Bases de Datos [7] [18]. A manera de ejemplo de la utilidad de estos sistemas, como caso práctico del uso de SQLfi, se han desarrollado tres aplicaciones concebidas para ser usadas vía WWW. Estas aplicaciones son las siguientes: sistema para la evaluación del desempeño de docentes y estudiantes [17], un sistema para la evaluación y promoción de empleados basado en méritos y trayectoria [1], concesionario virtual de autos que permite compras ajustadas a preferencias del usuario [6], sistema de apoyo al control de proyectos que aprovecha las experiencias previas [5], herramienta para ayuda en la elaboración de evaluaciones de cursos académicos [14], sistema de apoyo al diagnóstico y tratamiento médico en enfermedades respiratorias comunes [4] y sistema de apoyo a la toma de decisión sobre personal académico basado en la encuesta de opinión estudiantil [8].

Aunque la mayoría de estos problemas o universos de discurso que hemos tratado han sido ampliamente estudiados y analizados en diversos contextos, la incorporación de la lógica difusa para la modelación de los datos no fue tarea sencilla. En primer lugar, nuestros programadores (estudiantes de licenciatura, ingeniería o tecnología a nivel universitario) desconocían los conceptos de esta teoría. Sin embargo, el tiempo de aprendizaje fue relativamente corto.

Las metodologías existentes para análisis y diseño de sistemas de información no fueron concebidas para sistemas de aplicaciones con requerimientos difusos, por lo cual, su uso constituyó la mayor dificultad en estas experiencias y evidenció la necesidad de crear herramientas metodológicas para atacar este problema. A pesar de los distintos esfuerzos que se han hecho para la incorporación de la lógica difusa en las bases de datos, no se han provisto de herramientas metodológicas para determinar cuándo una aplicación informática

requiere del uso de esta lógica. De allí la motivación del presente trabajo cuyo aporte novedoso va justamente en el sentido de proveer de estas herramientas.

De acuerdo con las investigaciones realizadas por los autores, el presente trabajo es el primero en establecer las bases para la determinación de la necesidad de uso de consultas difusas para el desarrollo de sistemas de aplicación reales. Para realizar este aporte, se estudiaron aspectos teóricos de lógica difusa, lingüística formal del lenguaje español y experiencias prácticas en el desarrollo de aplicaciones que usan un sistema de consultas difusas a bases de datos realiconales.

El resto de este artículo se ha estructurado como sigue: — La sección II trata sobre los Conceptos de Lógica Difusa en SQLf; — La sección III se concentra en determinar los Elementos Gramaticales en Requerimientos Difusos; — La sección IV presenta Aplicaciones de SQLf Estudiadas para la realización de esta investigación; — La sección V presenta la Definición de Características que identifican a las aplicaciones que requieren el uso de un sistema de consultas difusas para su implementación y evidencia la presencia de tales características en las aplicaciones estudiadas; — La sección VI pone en relieve las Conclusiones a que se ha llegado con este trabajo; y, finalmente, — La sección VII apunta hacia los Trabajos Futuros que esperamos realizar dando continuidad a la investigación aquí reportada.

II. Conceptos de Lógica Difusa en SQLf

Los conjuntos difusos fueron introducidos en 1965 por Zadeh[29], su intención era modelar clases imprecisas en Sistemas de Control. Con el tiempo, los conjuntos difusos han sido usado en una gran variedad de aplicaciones, fundamentalmente en el Área de Inteligencia Artificial, sin embargo su aplicabilidad en Bases de Datos y Sistemas de Información no ha sido aún completamente explotada en la práctica, a pesar de los diversos esfuerzos teóricos que se han realizado en este sentido en las últimas dos décadas [13] [25] [27]. En un conjunto difuso, cada elemento está provisto de un grado que representa su membresía al conjunto difuso. Estos grados inducen un orden que define preferencias. La membresía es entonces definida como una función cuyo rango es el intervalo real $[0,1]$. La función de membresía de un conjunto difuso F es denotada con el símbolo μ_F .

Algunos operadores especiales permiten establecer conexiones entre conjuntos difusos y conjuntos clásicos. El operador llamado *support*, define el conjunto que contiene todos los elementos que no están completamente excluidos del conjunto difuso, es decir, aquellos cuyo grado de membresía es distinto de cero. El operador llamado *core*, define el conjunto que contiene todos los elementos que completamente incluidos en el conjunto difuso, es decir, aquellos cuyo grado de membresía es uno. El operador llamado α -*cut*, define el conjunto que contiene todos los elementos cuya membresía en el conjunto difuso es mayor o igual que el nivel $\alpha \in [0,1]$. El operador llamado *strict- α -cut*, define el conjunto que contiene todos los elementos cuya membresía en el conjunto difuso es mayor estricto que el nivel $\alpha \in [0,1]$.

La teoría de Conjuntos Difusos es la base de la Lógica Difusa. En esta lógica, el valor de verdad de una condición (o grado de satisfacción) está en el intervalo real $[0,1]$. El valor 0 se entiende como "completamente falso", y el valor 1 es "completamente cierto". El valor de verdad de una proposición "s" se denota como $\mu(s)$.

La Lógica Difusa puede ser usada para especificar condiciones de búsqueda en SQLf. Éste es un lenguaje resultante de extender SQL con la capacidad de usar una condición en lógica difusa en cualquier lugar donde SQL clásico permite el uso de una condición Booleana. SQLf provee todas las estructuras de consulta de SQL según las normas estándares de 1986, 1992 y 1999. Los diversos tipos de consultas son naturalmente extendidos al permitir condiciones difusas. La estructura más básica de consulta en SQLf es el bloque multirelacional, cuya sintaxis y semántica es la siguiente::

```
SELECT <attributes> FROM <relations> WHERE
<Fuzzy Conditions> WITH CALIBRATION [k| $\alpha$ |k, $\alpha$ ]
```

El resultado de esta consulta es el conjunto difuso de filas con los atributos proyectados de la cláusula SELECT en el producto cartesiano de las relaciones en la cláusula FROM que satisfacen la condición difusa de la cláusula WHERE. Como es un conjunto difuso, cada elemento está dotado de un grado de satisfacción y los elementos cuyo grado es cero no forman parte de la respuesta. Las condiciones difusas pueden involucrar términos definidos por el usuario, operadores y/o subconsultas difusas. La cláusula WITH CALIBRATION es opcional, ésta indica la escogencia de las mejores respuestas. Se han propuesto dos tipos de calibraciones: — Calibración Cuantitativa, que indica la escogencia de las mejores k respuesta, de acuerdo

a su grado de satisfacción. — Calibración Cualitativa que indica la escogencia de la respuestas cuyo grado de pertenencia es mayor o igual a un nivel mínimo de satisfacción especificado α .

Las condiciones difusas involucran términos lingüísticos que expresan preferencias de usuario. Por ello se hace necesario proveer de un lenguaje para la definición de estos elementos del lenguaje de la Lógica Difusa. Con tal fin fue concebido el sublenguaje de definición de datos SQLf-DDL [26]. Éste presenta construcciones adecuadas para la especificación de las distintas clases de términos lingüísticos [30] [2].

A. Predicados Difusos

Predicados, que son los componentes atómicos de la lógica difusa, definidos por una función de membresía (o conjunto difuso). Éstos corresponden a la clase de términos que se conocen en la literatura como “etiquetas lingüísticas”, sin embargo en este trabajo, como en todos los trabajos previos sobre SQLf, aquí se denominarán “predicados difusos”. La forma más obvia de definir estos predicados es por extensión, listando cada uno de los elementos cuyos grados son diferentes de cero, indicando el grado respectivo. Una forma que podría usarse en dominios numéricos es el dar una expresión aritmética para el cálculo del grado. Finalmente, la más común es definirlo mediante una función de membresía de forma trapezoidal para lo cual sólo se requiere especificar los cuatro valores en un dominio continuo que determinan el *support* y el *core* del conjunto difuso.

Los predicados difusos se definen en SQLf mediante la sentencia:

```
CREATE FUZZY PREDICATE <nombre> ON <dominio> AS <cjto difuso>
```

Donde: — <dominio> es o bien un rango de caracteres, o un rango de enteros, o un rango de números reales, o bien o un tipo escalar definido por el usuario; — <cjto difuso> es una especificación de conjunto difuso, que puede ser de una de las siguientes tres formas: — una distribución de posibilidades representada por un trapecoide con la sintaxis (<valor1>, <valor2>, <valor3>, <valor4>) pudiendo usarse el la palabra clave *INFINITE* en lugar de un valor cuya semántica es infinito; — una distribución de posibilidades representada por extensión, con la sintaxis: {<grado₁>/<valor₁>, ..., <grado_n>/<valor_n>}; — una expresión aritmética que usa la variable predefinida *x* que denota el argumento del predicado.

Por ejemplo, la siguiente sentencia en SQLf podría usarse para especificar el predicado difuso *joven* según la interpretación dada por algún usuario. Los grados de satisfacción de este predicado serían los que arroja la función de membresía graficada en Fig. 1.

```
CREATE FUZZY PREDICATE joven ON 0..120 AS (INFINITE, INFINITE, 25, 50)
```

Veamos otro sencillo ejemplo: la siguiente sentencia en SQLf especifica el predicado difuso de los colores *favoritos* según la interpretación dada por algún usuario. En este caso, los grados de satisfacción de este predicado son dados explícitamente. Los elementos del conjunto de colores que no se encuentran en la lista, tendrían grado de satisfacción 0 para este predicado.

```
CREATE FUZZY PREDICATE favorito ON colores AS {1/verde, 0.7/azul, 0.7/rojo, 0.5/blanco}
```

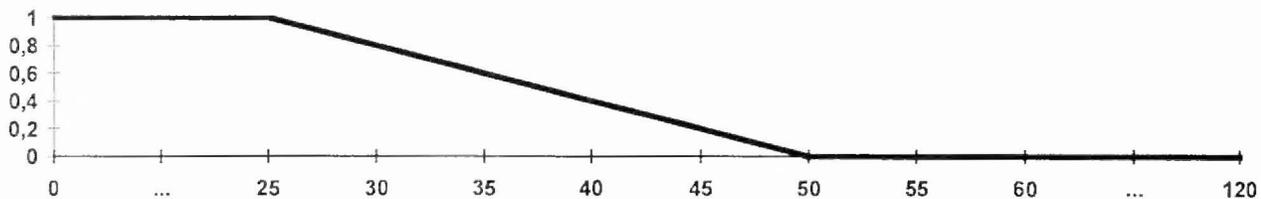


Fig. 1 Función de Membresía de una posible interpretación del predicado difuso joven

B. Modificadores Difusos

Modificadores son términos que permiten definir predicados modificados por medio de transformaciones sobre funciones de membresía. Típicamente estas transformaciones pueden intensificar, relajar, desplazar o invertir el concepto asociado. Un modificador intensifica cuando su efecto es disminuir el grado de satisfacción de la proposición. Es un relajador cuando al aplicarlo el grado de satisfacción de la proposición lógica aumenta. Un desplazamiento consiste en que el grado es el que originalmente tendría si el dato estuviera desplazado en su dimensión. Un modificador puede invertir un predicado al hacer una imagen especular de la función de membresía por un eje central vertical u horizontal.

Los modificadores difusos se definen en SQLf mediante la sentencia:

```
CREATE FUZZY MODIFIER <nombre> AS <transformación>
```

Donde *<transformación>* es una especificación de uno de estos tres tipos: — *TRANSLATION <d>*, con *<d>* es un desplazamiento, es decir, un valor que se suma al argumento del predicado original para hacer la traslación: $\mu_{modP}(x) = \mu_P(x+d)$; — *POWER <n>*, en este caso el predicado modificado *modP* es definido como: $\mu_{modP}(x) = (\mu_P(x))^n$; — *<θ> POWER <n>*, siendo *<θ>* una t-norma¹ o s-norma² que se especifica mediante una expresión con las variables predefinidas para el argumento derecho *x*, y para el argumento izquierdo *y*, en este caso el predicado *modP* es definido como:

$$\mu_{modP}(x) = \underbrace{(P\theta \dots \theta P)}_{n \text{ veces } P}(x).$$

Por ejemplo, el modificador *extremadamente* podría ser definido en SQLf, de la siguiente manera:

```
CREATE FUZZY MODIFIER extremadamente AS max(x+y-1,0) POWER 4
```

Nótese que el 0 en la expresión *max(x+y-1,0)* actúa como una cota para el caso en que la suma *x+y* sea menor que 1 el resultado final no se salga del intervalo real $[0, 1]$ en el cual se define las normas triangulares.

1 operador binario cerrado en $[0,1]$, conmutativo, asociativo, con elemento neutro 1

2 operador binario cerrado en $[0,1]$, conmutativo, asociativo, con elemento neutro 0

C. Comparadores Difusos

Comparadores, una clase de predicados difusos definido sobre pares de elementos, ellos establecen una comparación difusa. Estas relaciones difusas se pueden definir por extensión o mediante el grado de satisfacción en un conjunto difuso para una distancia entre los elementos comparados, la cual puede ser la diferencia o el cociente.

Estos comparadores se crean mediante la sentencia

```
CREATE COMPARATOR <símbolo> ON <dominio> AS <par> IN <cjto difuso>
```

Donde *<símbolo>* es el identificador del comparador, el cual puede ser un alfanumérico o simbólico, *<dominio>* es o bien un rango o tipo de caracteres o números, o bien un tipo escalar definido por el usuario, *<par>* es la expresión que permite calcular el valor de la comparación, puede ser *(x,v)* para un par ordenado, *(x-y)* para una diferencia o *(x/y)* para, la variable *x* denota el término izquierdo; y la *y* el derecho; *<cjto difuso>* es la especificación de una función de membresía así como las de los predicados difusos.

Por ejemplo, en SQLf se podría definir el comparador *>>* (mucho mayor que) así:

```
CREATE COMPARATOR >> ON NUMBER AS (x/y) IN (1,10,INFINITE,INFINITE)
```

D. Conectores Difusos

Conectores, operadores definidos para la combinación de condiciones difusas. La negación, conjunción y disyunción difusas son extensiones de sus equivalentes clásicas. Preservan la correspondencia existente con los operadores de conjunto: complemento, intersección y unión. El lenguaje SQLf permite crear conectores mediante la sentencia:

```
CREATE FUZZY CONNECTOR <nombre> AS <expresión>
```

Donde *<expresión>* es la expresión que permite calcular el valor del predicado compuesto a partir de los valores de sus miembros. En esta expresión el grado de satisfacción de la condición a la izquierda se denota con el símbolo *x* mientras que el del miembro derecho lo representa el símbolo *y*.

Por ejemplo, un usuario podría querer usar un conector para la implicación

```
CREATE FUZZY CONNECTOR imp AS max(1-x,y).
```

E. Cuantificadores Difusos

Cuantificadores, términos que describen cantidades imprecisas. Éstos son una extensión de los cuantificadores usuales universal y existencial. Se clasifican según su naturaleza en “absolutos” y “proporcionales”. Los absolutos se definen sobre los números reales, pues describen cantidades difusas, mientras que los proporcionales se definen sobre el intervalo [0,1], pues éstos tienen la intención de describir cantidades relativas a un todo, por lo que son fraccionarias. Los cuantificadores difusos son interpretados mediante conjuntos difusos definidos con funciones de membresía trapezoidal. Según su comportamiento se clasifican en “crecientes”, “decrecientes” y “unimodales”.

Para ello se provee de la construcción sintáctica

```
CREATE <naturaleza> FUZZY QUANTIFIER <nombre> AS <valor1,valor2,valor2,valor4>
```

Donde <naturaleza> puede ser *ABSOLUTE* o *PROPORTIONAL*

Por ejemplo, de acuerdo con las preferencias de un usuario, se podría definir un cuantificador difuso expresando una cantidad “entre 20 y 30” y otro expresando la idea de “la mayoría”, como sigue:

```
CREATE ABSOLUTE FUZZY QUANTIFIER Entre20y30 AS (15,20,30,35)
```

```
CREATE PROPORTIONAL FUZZY QUANTIFIER LaMayoría AS (0.5,0.7,INFINITE,INFINITE)
```

III. Elementos Gramaticales en Requerimientos Difusos

Las consultas difusas tienen como propósito el dar respuesta a requerimientos de usuarios que no son fácilmente expresables con condiciones lógicas clásicas debido a su naturaleza vaga, frecuentemente dependiente del contexto y de la percepción o preferencias de los usuarios.

Nuestro aporte en el estudio de los elementos gramaticales del lenguaje es que el analista de sistema al determinar los requerimientos de información de una aplicación podrá detectar algunos elementos gramaticales que son indicadores de su naturaleza vaga. Este análisis se haría en base a la expresión de los requerimientos del sistema en el lenguaje natural. Cada uno de estos elementos gramaticales de naturaleza vaga tienen su representación en lógica difusa de manera que los requerimientos podrían ser implementados mediante consultas difusas en SQLf. En este trabajo

nosotros ponemos en manifiesto cuál o cuáles serían los tipos de términos difusos correspondientes a los elementos gramaticales, de manera que proveemos una herramienta al desarrollador del sistema que la ayudará en el diseño de las consultas difusas a la base de datos.

Hicimos un estudio de los diferentes elementos gramaticales del idioma Español [24], como resultado de este estudio, hemos determinando cuáles de ellos son indicadores de vaguedad. Para cada tipo de elemento gramatical determinamos de qué manera pueden estos ser representados en lógica difusa. Concluimos que diversas formas de adjetivos y adverbios son los elementos gramaticales de la vaguedad en la expresión del lenguaje natural que nos ayudan a determinar requerimientos difusos. La función gramatical de los adjetivos es acompañar y modifican al sustantivo. Los adverbios son palabras que básicamente modifican al verbo, pero también pueden aparecer acompañando un adjetivo u otro adverbio.

A. Adjetivos Calificativos de Grado Positivo

Los adjetivos calificativos son llamados así porque añaden una nota de cualidad. Estos adjetivos pueden tener un grado de intensidad asociado: *positivo*, *comparativo*, *superlativo relativo* o *superlativo absoluto*. El grado *positivo* corresponde al adjetivo en su forma original. Ejemplos de esta clase de adjetivos son las palabras: *bueno*, *malo*, *grande*, *pequeño*, *alto*, *bajo*, *externo*, *interno*, *joven*, *viejo*, *largo*, *corto*, *barato*, *caro*, *eficaz*, *productivo*, *nuevo*, *antiguo*, *pobre*, *rico*, *simple*, *complejo*, *pesado* y *liviano*. Los adjetivos calificativos de grado positivo pueden indicar términos lingüísticos usados como predicados difusos en el contexto de un requerimiento vago de un sistema de información.

B. Adjetivos Calificativos de Grado Comparativo

El grado comparativo se expresa normalmente con los adverbios *más* y *menos* acompañando a un adjetivo calificativo de grado positivo, indicando superioridad o inferioridad. Por ejemplo *más joven*, *menos productivo*, *más eficaz*. También hay adjetivos calificativos de grado comparativos puro, como lo son las palabras: *mejor*, *peor*, *mayor*, *menor*, *superior*, *inferior*, *exterior* e *interior*. En caso de observar que en la formulación en lenguaje natural de un requerimiento aparezca un adjetivo de este tipo, esto podría estar indicando

que el requerimiento es vago y que se necesita de un comparador difuso para expresarlo.

C. Adjetivos Calificativos de Grado Superlativo Relativo

El grado superlativo relativo para un adjetivo se marca añadiendo simplemente un artículo determinado al adjetivo en su grado comparativo. Este modo tiene dos versiones, una de superioridad y otra de inferioridad. Por ejemplo, la oración siguiente presenta los dos modos: "El más productivo empleado de la empresa es el menos antiguo de su departamento". Otros adjetivos calificativos de grado superlativo relativo son las palabras: *óptimo, pésimo, máximo, mínimo, supremo, sumo, infimo, extremo e íntimo*. El grado superlativo relativo implica una comparación con otros miembros del mismo conjunto (dominio o universo). El uso de este tipo de adjetivos también podría ser indicio de estar ante un requerimiento vago el cual debe usar una combinación de cuantificador y comparador difuso para su expresión como una consulta.

D. Adjetivos Calificativos de Grado Superlativo Absoluto

Las terminaciones *ísimo* y *érrimo* caracterizan a los adjetivos calificativos de grado superlativo absoluto, los cuales implican una comparación con un modelo universal (ideal). Ejemplos de este tipo de adjetivo son: *buenísimo, malísimo, novísimo, viejísimo, antiquísimo, grandísimo, pequeñísimo, altísimo, bajísimo, riquísimo, y paupérrimo*. Este tipo de elementos gramaticales tiene sus equivalente mediante el uso del adverbio *muy* seguido de un adjetivo calificativo de grado positivo. Si un adjetivo calificativo de grado superlativo absoluto aparece en la expresión de un requerimiento en lenguaje natural, este podría ser representado en lógica difusa mediante o un predicado difuso o un predicado con un modificador difuso.

E. Adjetivos Determinativos indefinidos cuantitativos

Los adjetivos determinativos añaden al sustantivo una nota que lo determina en algún sentido. Entre ellos nos interesan los indefinidos que añaden una nota vaga. En particular, nos enfocamos en los cuantitativos, tales como: *algunos, bastantes, cuantiosos, demasiados, muchos, múltiples, ningunos, numerosos, pocos, tantos, todos, unos y varios*. Obviamente estos

adjetivos son cuantificadores lingüísticos que podrían tener una interpretación difusa, de manera que su uso en la expresión de un requerimiento lo caracteriza como vago.

F. Adverbios Variables

Los adverbios básicamente modifican al verbo, pero también pueden aparecer acompañando un adjetivo u otro adverbio. Entre los adverbios que nos interesan están los variables, como: *bien, mal, grandemente, pequeñamente, altamente, bajamente, externamente internamente, óptimamente, pésimamente, máximamente, mínimamente, supremamente, sumamente, ínfimamente y extremamente*. Estos adverbios pueden ser modelados como modificadores difusos para la expresión de consultas. Otros ejemplos son las palabras *cerca* y *lejos*. Estas últimas son claramente representantes los comparadores difusos. De manera que el uso de adverbios variables es un indicativo de la posible vaguedad del requerimiento.

G. Adverbios Invariables de cantidad

Los adverbios invariables son unos que no tienen grados. Hay una gran variedad de ellos, entre los cuales nos interesan los adverbios invariables de cantidad. Ejemplos de este tipo de adverbios lo son las palabras: *bastante, harto, medio, muy, poco, mucho, tanto, tan, casi y algo*. Estos adverbios pueden ser interpretados como modificadores difusos, de manera que su uso en cualquier requerimiento de un sistema de información, pondría en evidencia la necesidad del uso de consultas difusas para su implementación.

IV. Aplicaciones de SQLf Estudiadas

En la Universidad Simón Bolívar, con la colaboración de estudiantes de la Ingeniería en Computación, así como estudiantes de Licenciatura en Computación provenientes de la Universidad Central de Venezuela, y mayormente con estudiantes de Tecnología en Informática del Instituto Universitario Tecnológico de la Región Capital Federico Rivero Palacios, se ha tenido la experiencia por más de cinco años en la construcción de aplicaciones que utilizan consultas difusas.

Los estudiantes que desarrollaron esta aplicación no tenían conocimiento previo de Lógica Difusa y su experiencia con SQL era la mínima adquirida en un primer curso básico de Bases de Datos. Ellos no

recibieron formación específica sobre la determinación de requerimientos difusos a través de elementos gramaticales, tal como se presenta en este artículo. Con este trasfondo, se desarrollaron aplicaciones que abarcan problemas de diferentes áreas de interés en que las consultas difusas podrían ser útiles.

Con el fin de poner en evidencia los requerimientos vagos y determinar las características que definen el tipo de aplicaciones que requieren del uso de un lenguaje como SQLf, para el presente estudio se tomaron siete de estas aplicaciones desarrolladas. En lo sucesivo se presenta una breve reseña de estas aplicaciones. Para la primera de ella se hace aquí un análisis de tres de los requerimientos de información que muestran el hecho de ser requerimientos vagos por el uso de algunos de los elementos gramaticales determinados en la sección anterior de este artículo.

A. Una Herramienta Web para la Evaluación de Desempeño Docente

Realizada por estudiantes de Licenciatura en Computación de la Universidad Central de Venezuela [17], es una aplicación que permite realizar encuestas sobre cursos y docentes, la cual busca apoyar el análisis de los resultados de dichas encuestas por medio de la discriminación de las respuestas y de la flexibilidad en el tipo de preguntas que se pueden efectuar en un sistemas de consultas a bases de datos. Se pueden hacer una gran variedad de consultas por parte del estudiante o del profesor. Este sistema da soporte a requerimientos de usuario como las siguientes, entre otros:

- *¿Cuáles preparadores son deficientes?*, en este requerimiento conseguimos el adjetivo calificativo de grado positivo *deficiente*. Éste claramente connota una noción vaga en el contexto del desempeño de los preparadores. Su modelación se haría mediante un predicado difuso.
- *¿Cuáles son los profesores con muy buen desempeño?*, se puede observar la frase compuesta del adverbio invariable de cantidad *muy* y el adjetivo calificativo en grado positivo *buen*. El adverbio *muy* es obviamente representable como un modificador difuso. Resulta también obvio que el adjetivo *buen* sería modelado como un predicado difuso.

- *¿Cuáles son los cursos de la facultad que tienen las mayores debilidades?*, aquí la frase *las mayores* es un adjetivo calificativo de grado superlativo relativo. Para su implementación es necesario que se defina un comparador difuso para el adjetivo *mayor* en el contexto de las debilidades de los cursos. También es necesario un cuantificador para la noción del adjetivo determinativo indefinido cuantitativo *todos* que está implícita en el grado superlativo relativo del adjetivo *los mayores*. Como en la lógica clásica existe el cuantificador universal que representa la noción de *todos* con una interpretación precisa, podría usarse este cuantificador o podría relajarse usando el adverbio invariable de cantidad *casi*, obteniendo en su lugar la forma adjetiva *casi todos* que sí sería evidentemente un cuantificador difuso.

B. Sistema de Promoción de Empleados

Fue realizado por estudiantes de Técnico Superior Universitario en Informática del Instituto Universitario de Tecnología de la Región Capital (IUT) [1]. El actor principal del sistema lo cumple el *Supervisor*, aunque posee también los actores de *Profesional* y *Administrador*. El *Supervisor* al ingresar al sistema puede colocar datos sobre evaluaciones de personal y llevar calificaciones referidas a los criterios de: Puntualidad, Responsabilidad, Relaciones Laborales, Productividad, Motivación, Liderazgo y Expresión. El supervisor puede buscar empleados que califiquen para ser promovidos en el cargo de su preferencia. Además el supervisor puede especificar su nivel mínimo de tolerancia (calibración cualitativa) y seleccionar un máximo de respuestas deseadas (calibración cuantitativa). Los criterios que involucra la selección de un candidato, corresponden a valores precisos o predicados difusos.

C. Sistema Web para Compra-Venta de Vehículos

Aplicación desarrollada por estudiantes de Informática del IUT [6]. En este sistema, un *Usuario* registrado puede ingresar y colocar una nueva oferta de vehículo. Los datos de los vehículos son los usuales que podrían conseguirse en los sitios de ventas virtuales de automóviles, tales como: fotos, marca, modelo, año de fabricación, potencia del motor, color, tipo de sistema de climatización, tipo de componente de

audio y entretenimiento a bordo, precio de la oferta, precio negociable o no, entre otros. Un *Usuario* puede buscar y seleccionar vehículos de las diversas ofertas publicadas mediante la combinación de varios criterios sobre los distintos elementos de datos que describen a un vehículo ofertado. La idea del sistema es que el usuario pueda hacer búsquedas basadas en preferencias sin necesidad de ser un experto en lógica difusa. El resultado de la búsqueda es una lista de los vehículos encontrados ordenada en forma descendente de acuerdo a la satisfacción de la consulta difusa. Los criterios de selección se basan en predicados difusos que están predefinidos en el sistema y otros que el usuario puede especificar mediante la interfaz.

D. Sistema de Evaluación de Consultas sobre Encuestas de Opinión Estudiantil

En la Universidad Simón Bolívar se han realizado por más de siete años encuestas para obtener la opinión de los estudiantes sobre las asignaturas que cursan y la manera en que las dictan sus profesores. Las encuestas tienen carácter confidencial y pretenden evaluar el desempeño docente a nivel individual y a nivel departamental. Los resultados de las mismas son entregados a los profesores y departamentos el trimestre siguiente a ser aplicadas. Este insumo sirve al departamento para evaluar el desempeño de sus docentes, así como también para programas de estímulo y promoción por parte de la universidad. Sin embargo, su importancia no ha sido explotada totalmente. La aplicación propuesta, realizada por estudiantes de Informática del IUT [8], formula consultas difusas a la base de datos de encuestas según las preferencias del profesor, jefes de departamentos, coordinadores de programas de estudios y/o comités de estímulo y promoción. Esta aplicación será muy útil en la toma de decisiones para la contratación, promoción y estímulo del personal académico.

E. Sistema para la Construcción de Evaluaciones Docentes

Desarrollado por estudiantes del IUT [14], es un sistema que pretende apoyar al docente en el diseño y preparación de las pruebas aplicadas a los estudiantes de un curso. Para ello, el sistema provee al docente de un conjunto de criterios para la preparación de pruebas o preguntas: dificultad, tiempo de elaboración, tiempo de corrección, entre otros. Estos criterios son imprecisos y subjetivos a la apreciación de cada docente.

Estas pruebas y/o preguntas son almacenadas en una base de datos que después puede ser consultada de acuerdo a las preferencias del usuario, así como también, a los grados y los resultados obtenidos en pruebas anteriores, por lo que los docentes pueden proyectar el desempeño esperado de sus estudiantes. El sistema también provee la generación de estadísticas basadas en requerimientos graduales definidos por los profesores.

F. Sistema de Diagnóstico Médico

Aunque se han desarrollado muchos sistemas de diagnóstico médico para diversas enfermedades, esta aplicación es una experiencia nueva en cuanto a su implementación en un sistema manejador de base de datos difusas. Dado que el campo médico es rico en términos imprecisos, que la bioinformática es uno de nuestros campos de interés y a la disposición de un especialista médico como soporte, se procedió a la implementación de una aplicación de este tipo con la ayuda de estudiantes de Informática del IUT [4]. La aplicación fue basada en el diagnóstico clínico de enfermedades respiratorias comunes a través del examen físico y otros exámenes adicionales fuera de la anatomía patológica. El diagnóstico se hace a través de la simulación de la decisión tomada por un médico especialista de acuerdo a técnicas y datos suplidos por éste, según su experiencia y conocimiento. El sistema provee la capacidad de listar diferentes diagnósticos con un grado de membresía asociado, facilidad para controlar la calidad del diagnóstico obtenido, la redefinición de los términos lingüísticos y la posibilidad de flexibilizar y parametrizar el diagnóstico obtenido.

G. Sistema para el Seguimiento de Control de Proyecto

Aplicación que abarca la administración de proyectos, tareas, recursos y costos, la selección de recursos técnicos y el liderazgo, así como también, el control y seguimiento del flujo de trabajo. Fue desarrollada por estudiantes de Informática del IUT [5], y pretende agregarle a este problema altamente conocido las ventajas de la lógica difusa para la formulación de requerimientos más ambiciosos. Las consultas difusas serán el soporte para obtener respuestas a preguntas como ¿cuál es el mejor equipo de trabajo para un proyecto con una fecha de entrega y presupuesto dados? ¿cuánto tiempo es requerido para liberar un prototipo funcional al cliente? ¿Qué acciones debería ejecutar un director de

proyecto cuando se han producido tiempos de retraso en la entrega a un cliente? Todas ellas constituyen un nivel gerencial del proyecto y requieren de flexibilidad y uso de criterios de preferencias parametrizables en las consultas. El sistema aporta una base de datos donde el líder del proyecto pueda aprovechar las experiencias para la toma de decisiones futuras.

V. Definición de Características

Luego de la experiencia con el desarrollo de las aplicaciones del SQLf que se estudiaron, pudimos concluir que existen características en común que permiten detectar si un problema o universo de discurso es susceptible a ser modelado usando lógica difusa. Estas características deben estar presentes en uno o más requerimientos a ser satisfechos por la base de datos del sistema en estudio. Las características detectadas fueron estas: *Intuitividad*, *Flexibilidad*, *Vaguedad*, *Tolerabilidad*, *Adaptabilidad*, *Gradualidad*, y *Gerencialidad*.

En este trabajo proponemos que el analista de sistema se verifique si la aplicación a desarrollar presenta estas características (todas o la mayoría de ellas), y de ser así, sin duda se está ante una aplicación que requiere del uso de un sistema de bases de datos con capacidad de consultas difusas.

En las subsecciones que se presentan a continuación, se definen y ejemplifican cada una de estas características tomando como base las aplicaciones estudiadas.

A. Intuitividad

Definimos *Intuitividad* como la cualidad de un sistema informático de ser intuitivo para los usuarios del mismo. Las consultas difusas se caracterizan por usar términos lingüísticos que son más cercanas al lenguaje natural que los valores o rangos y formulaciones complejas que se aproximan a ellas pero con lenguajes de consultas clásicas. Ellas debe ser fácilmente entendibles y manejables por el usuario final.

Tabla 1. Intuitividad en algunos aspectos de las aplicaciones de estudiadas

Herramienta de Evaluación de Cursos y DOCentes basada en consultas difusas	Se incluyen consultas como ¿Cuáles son los profesores con mejor desempeño?, ¿Cuáles preparadores son deficientes? o ¿Cuáles son los cursos que tienen más debilidades?
Sistema de promoción de empleados	Se proveen de mecanismos de especificación visual que ayudan al usuario final en la configuración de la consulta involucrando características del perfil del empleado buscado.
Sistema de compra de vehículos	Se provee una interfaz visual para especificar preferencias en cuanto a aspectos como la potencia, el modelo, la marca, el color. Las interfaces se asemejan a los sistemas clásicos de compra-venta de autos.
Sistema de evaluación de consultas sobre encuestas de opinión estudiantil	Las encuestas son realizadas como una lista de aptitudes y hechos que los estudiantes pueden evaluar. Las interfaces se diseñaron para que el usuario final pueda visualizar de manera gráfica los términos difusos
Sistema para la construcción de evaluaciones docentes	Se provee de un ambiente agradable e intuitivo al profesor para preparar sus exámenes, usando preguntas que ha registrado previamente y estadísticas sobre sus resultados.
Sistema de diagnóstico médico	Se proveen patrones sencillos y flexibles para la formulación de las consultas pertinentes al área de enfermedades respiratorias
Sistema para el seguimiento de control de proyecto	Responde preguntas como ¿cuál es el mejor grupo de programadores para una tarea?, ¿cuánto tiempo es requerido para producir un prototipo funcional?, entre otros

B. Flexibilidad

La *Flexibilidad* la definimos en este contexto como la capacidad del sistema o aplicación de incluir en las respuestas valores en los bordes que pueden ser de interés para el usuario de acuerdo a su preferencia aunque no cumplan de una manera rígida con el ideal buscado. Es decir, el resultado de las consultas difusas puede incluir respuestas que serían rechazadas por una consulta precisa.

Tabla 2. Flexibilidad en algunos aspectos de las aplicaciones de estudiadas

Herramienta de Evaluación de Cursos y DOCentes basada en consultas difusas	Las definiciones de los términos lingüísticos se realizan a través de conjuntos difusos
Sistema de promoción de empleados	La interfaz gráfica permite al usuario especificar sus preferencias. Los criterios involucran valores precisos y predicados difusos
Sistema de compra de vehículos	Se manejan tanto valores precisos como difusos para los diversos criterios de búsqueda

Sistema de evaluación de consultas sobre encuestas de opinión estudiantil	Los usuarios pueden definir los términos lingüísticos y disponen de un conjunto predefinido que pueden redefinir si lo desean
Sistema para la construcción de evaluaciones docentes	Se proveen evaluaciones preexistentes las cuales el profesor puede adaptar de acuerdo a sus preferencias
Sistema de diagnóstico médico	Se manejan términos precisos y difusos en las consultas. Se incluyen posibles diagnósticos que en otro contexto serían rechazados
Sistema para el seguimiento de control de proyecto	Las definiciones de los términos lingüísticos se realizan a través de conjuntos difusos.

C. Vaguedad

La imprecisión y/o incertidumbre está presente en sistemas de conocimiento y de razonamiento humano. Estos conceptos son englobados aquí bajo el término de *Vaguedad*. Con esto entendemos que los sistemas abarcan consultas cuyas proposiciones pueden ser parcialmente ciertas o parcialmente falsas. La información manejada puede estar caracterizada por la incertidumbre. Lo cual incluye datos imprecisos o consultas vagas que reflejan preferencias de los usuarios.

Tabla 3. Vaguedad en algunos aspectos de las aplicaciones de estudiadas

Herramienta de Evaluación de Cursos y DOCentes basada en consultas difusas	Se manejan términos como "bueno", "regular", "deficiente" para la clasificación de notas y estudiantes. Se representan cuantificadores del tipo "la mayoría"
Sistema de promoción de empleados	Se manejan predicados difusos como "pocos" y "muchos" para la antigüedad, la experiencia previa y los cursos realizados
Sistema de compra de vehículos	Se incluyen términos como "alto", "medio" y "bajo" para el precio, la potencia, etc
Sistema de evaluación de consultas sobre encuestas de opinión estudiantil	Se manejan predicados difusos como "bueno", "excelente", "regular", "deficiente" comparadores difusos: "MuchoPeor", "Peor", "MuchoMejor" y "Mejor", cuantificadores difusos "LaMayoría", "LaMitad" y "LaMinoría"
Sistema para la construcción de evaluaciones docentes	Se dispone de predicados difusos como "bajo", "moderado", "alto" y "muy alto" para las variables como Dificultad y Tiempo de Corrección
Sistema de diagnóstico médico	Se manejan términos difusos como "severo", "moderado" o "bajo", para Intensidad y "agudo" o "crónico" para Duración
Sistema para el seguimiento de control de proyecto	Se incluyen términos difusos como "corta", "media" y "larga" para la Duración; "poco", "suficiente" y "mucho" para el Presupuesto

D. Tolerabilidad

Para nuestros propósitos, definimos la *Tolerabilidad* como la capacidad de establecer niveles de tolerancia dentro de la imprecisión, estableciendo así rangos aceptables del grado de satisfacción de las respuestas obtenidas por el sistema. Esto es lo que en SQLf se conoce como la calibración. Aunque las consultas de nuestro interés se caracterizan por su flexibilidad e imprecisión, éstas podrían estar limitadas por un cierto grado que evita la generación de respuestas superpuestas y/o respuestas indeseadas por su bajo nivel de satisfacción o calidad.

Tabla 4. Tolerabilidad en algunos aspectos de las aplicaciones de estudiadas

Herramienta de Evaluación de Cursos y DOCentes basada en consultas difusas	Se coloca la cláusula "WITH CALIBRATION" a las consultas para garantizar la calidad
Sistema de promoción de empleados	El supervisor puede especificar gráficamente su nivel mínimo de tolerancia (calibración cualitativa) y también seleccionar el máximo de respuestas deseadas (calibración cuantitativa)
Sistema de compra de vehículos	El usuario especifica la "precisión de la búsqueda", (calibración cualitativa) y la "cantidad de resultados" (calibración cuantitativa)
Sistema de evaluación de consultas sobre encuestas de opinión estudiantil	La consulta difusa puede ser calibrada de acuerdo al grado de satisfacción, tanto por calidad como por cantidad de respuestas.
Sistema para la construcción de evaluaciones docentes	Se coloca la cláusula "WITH CALIBRATION" a las consultas que el usuario requiera
Sistema de diagnóstico médico	Se proveen patrones sencillos y flexibles para la formulación de las consultas pertinentes al área de enfermedades respiratorias
Sistema para el seguimiento de control de proyecto	Se proveen mecanismos para calibrar la consulta difusa

E. Adaptabilidad

Se requiere del manejo de las preferencias del usuario en las consultas que sean parametrizables, a esto lo denominamos *Adaptabilidad*. Es decir, cada término lingüístico debe ser susceptible a cambios debido a percepciones de los diferentes usuarios o contextos.

Tabla 5. Adaptabilidad en algunos aspectos de las aplicaciones de estudiadas

Herramienta de Evaluación de Cursos y DOCentes basada en consultas difusas	Las definiciones de los términos difusos pueden ser cambiados por el administrador del sistema
Sistema de promoción de empleados	La interpretación de los términos difusos la da el usuario según sus preferencias en el momento de hacer la consulta

Sistema de compra de vehículos	Algunos criterios por su complejidad son predefinidos otros pueden ser cambiados de acuerdo a las preferencias del usuario
Sistema de evaluación de consultas sobre encuestas de opinión estudiantil	Los usuarios pueden redefinir sus preferencias de acuerdo a los parámetros que necesite el evaluador
Sistema para la construcción de evaluaciones docentes	Los profesores pueden adaptar los términos difusos a sus necesidades
Sistema de diagnóstico médico	Permite al médico redefinir los términos difusos de acuerdo a sus preferencias, basadas en conocimiento, experiencia y percepción profesional.
Sistema para el seguimiento de control de proyecto	Se permite modificar las definiciones de los términos difusos de acuerdo a las preferencias del usuario

F. Gradualidad

Entendemos por *Gradualidad* el hecho que las respuestas a los requerimientos de las aplicaciones sean discriminadas por grados de acuerdo con la satisfacción de las condiciones involucradas. Cada elemento o registro en el conjunto de respuestas resultantes de una consulta tiene asociado un valor que expresa su importancia o preferencia para el usuario de acuerdo al requerimiento expresado.

Tabla 6. Gradualidad en algunos aspectos de las aplicaciones de estudiadas

Herramienta de Evaluación de Cursos y DOCentes basada en consultas difusas	Los resultados de las consultas difusas se muestran en pantalla, indicando, para cada elemento en la respuesta, su grado de satisfacción a la consulta difusa
Sistema de promoción de empleados	El resultado es ordenado en forma decreciente por el grado de satisfacción
Sistema de compra de vehículos	El resultado se ordena de acuerdo al grado de satisfacción de la consulta difusa
Sistema de evaluación de consultas sobre encuestas de opinión estudiantil	El grado de satisfacción de cada registro se incluye en la respuesta de la consulta.
Sistema para la construcción de evaluaciones docentes	El grado de satisfacción se incluye en las respuestas a las consultas difusas, como una columna adicional en la salida
Sistema de diagnóstico médico	Se provee de una lista de diagnósticos aproximados con diferentes grados de membresía
Sistema para el seguimiento de control de proyecto	Se incluye una columna "Precisión" en la respuesta a las consultas que indica el grado de satisfacción

G. Gerencialidad

Aunque las consultas difusas pueden ser útiles para personas en cualquier nivel de la organización o para usuarios comunes, ellas toman mayor aplicabilidad e importancia en el entorno gerencial, donde los términos difusos aparecen con mayor frecuencia. Este último

punto es apoyado por el principio de incompatibilidad de Zadeh [1975], el cual afirma que en la medida que crece la complejidad de un sistema en esa misma medida decrece la capacidad de escribir enunciados precisos sobre su funcionalidad. Hemos notado que la lógica difusa es más aplicable cuando el rol del usuario final se encuentra en un nivel gerencial. Por ello definimos en concepto de *Gerencialidad* como la cualidad de un sistema de dar soporte a la toma de decisiones.

Tabla 7. Gerencialidad en algunos aspectos de las aplicaciones de estudiadas

Herramienta de Evaluación de Cursos y DOCentes basada en consultas difusas	El usuario con perfil de profesor tiene acceso a consultas para el análisis de resultados de la encuesta. Las respuestas son ordenadas por criterios de preferencia
Sistema de promoción de empleados	El supervisor puede buscar empleados que califiquen para ser promovidos en el cargo de su preferencia. También puede seleccionar "preferidos" y acceder su hoja de vida para ver detalles.
Sistema de compra de vehículos	El usuario puede ir al detalle del vehículo y retornar al resultado de la búsqueda cuando lo desee
Sistema de evaluación de consultas sobre encuestas de opinión estudiantil	Los resultados de las consultas son insumos importantes para jefes de departamentos y coordinadores de carreras, con el fin de evaluar el desempeño de cursos y profesores, también para programas de promoción y estímulo
Sistema para la construcción de evaluaciones docentes	El sistema sugiere configuraciones de examen de acuerdo con las preferencias del profesor quien finalmente escoge entre las propuestas.
Sistema de diagnóstico médico	Se incorporan aspectos sobre la relación Síntoma-Enfermedad, Signos-Enfermedad y Historia-Enfermedad para ayudar a producir un diagnóstico
Sistema para el seguimiento de control de proyecto	Las funciones de administración y selección de recursos, así como el control del flujo de trabajo, provistas por este sistema, son requeridas en la toma de decisiones

VI. Conclusiones

En las últimas dos décadas se han hecho varios esfuerzos por dotar a los sistemas de bases de datos con nuevas capacidades basadas en el uso de conjuntos difusos para la representación y manipulación de vaguedad en consultas y datos. Sin embargo esto no se ha reflejado en el desarrollo de sistemas de información donde las metodologías de desarrollo existentes no fueron concebidas para sistemas de aplicaciones con

requerimientos difusos. El aporte de este trabajo es el establecer las bases para la determinación de la necesidad de uso de un manejador de bases de datos difusas en el desarrollo de aplicaciones, lo cual podría ser muy provechoso para una nueva generación de sistemas de información basados en preferencias de usuarios que den soporte a la toma de decisiones. También se dan lineamientos claros que ayudarán al desarrollador para la modelación de requerimientos difusos.

Se estudiaron los conceptos fundamentales de la lógica difusa presentes en el lenguaje flexible para consultas difusas a bases de datos relacionales SQLf, los elementos gramaticales del lenguaje natural y siete sistemas de información experimentales desarrolladas sobre un servidor de consultas difusas.

Los aplicaciones estudiados fueron: "Herramienta Web para la Evaluación de Desempeño Docente", "Sistema de Promoción de Empleados", "Sistema Web para Compra-Venta de Vehículos Usados", "Sistema de Consultas sobre Encuestas de Opinión Estudiantil", "Sistema para la Construcción de Evaluaciones Docentes", "Sistema de Diagnóstico Médico" y "Sistema para el Seguimiento de Control de Proyecto". Estas aplicaciones fueron desarrolladas por diferentes equipos de estudiantes de informática, en su mayoría en nivel de técnico superior universitario.

El estudio muestra que las aplicaciones que usan la lógica difusa pueden estar dirigidas a usuarios comunes sin conocimiento previo de la teoría subyacente. Asimismo se observó que mientras mayor sea el nivel de gerencia o toma de decisión de un usuario, se hace mas beneficioso el uso de la lógica difusa: "mientras mas complejo es un sistema, mas difícil es de expresar en términos precisos".

Se definieron siete características fundamentales que debe presentar un sistema de información para que su automatización requiera el uso de un manejador de bases de datos difusas, estas son: *Intuitividad, Flexibilidad, Vaguedad, Tolerabilidad, Adaptabilidad, Gradualidad, y Gerencialidad.*

Se detectaron siete clases de elementos gramaticales del lenguaje natural que, al usarse en la expresión de los requerimientos de una aplicación, indican la necesidad del uso de la lógica difusa para dar una expresión adecuada a la consulta dentro de un sistema automatizado. A cada una de estas clase de elementos gramaticales se le asocia un tipo (o dos, en algunos casos) de término difuso que lo puede representar, a saber: los adjetivos calificativos de grado positivo

(predicados difusos), los adjetivos calificativos de grado comparativo (comparadores difusos), los adjetivos calificativos de grado superlativo relativo (cuantificador y comparador difuso), los adjetivos calificativos de grado superlativo absoluto (predicados o modificadores difusos, según el caso), los adjetivos determinativos indefinidos cuantitativos (cuantificadores difusos), los adverbios variables (comparadores o modificadores difusos), los adverbios invariables de cantidad (modificadores difusos).

VII. Trabajos Futuros

Estamos extendiendo las metodologías de Análisis de Requerimientos para Bases de Datos con la intención de incorporar los requerimientos difusos de información de acuerdo con los resultados de la investigación aquí reportada. Estudiaremos a futuro las Metodología de Desarrollo de Software a fin de identificar las extensiones que fueren necesarias para aplicaciones con requerimientos difusos. Pretendemos crear marcos de desarrollo para agilizar la construcción de este tipo de aplicaciones usando el lenguaje Java y el sistema de consultas difusa SQLfi. Se ha emprendido una campaña nacional de difusión de la tecnología de bases de datos difusa con el fin de determinar y desarrollar nuevas aplicaciones y que éstas sean ampliamente usadas en nuestro entorno de influencia.

VIII. Agradecimientos

Agradecemos primeramente a Aquél quien es Nuestra Luz y Nuestro Guía, de quien proviene la ciencia, la sabiduría y la inteligencia, que nos ha dotado de todo cuantos somos y cuanto tenemos, a Nuestro Amigo Fiel, Jesucristo Nuestro Señor: Sin Él nada podemos hacer. También reconocemos el valioso aporte del Fondo Nacional Venezolano para la Ciencia, Innovación y Tecnología FONACIT al apoyarnos financieramente a través de la subvención G-2005000278. Finalmente, damos las gracias a todos nuestros colaboradores: "Los Chicos del IUT" quienes diseñaron y programaron las aplicaciones, y las profesoras Marlene Goncalves de la Universidad Simón Bolívar y Livia Borjas del IUT Federico Rivero Palacios, quienes gentilmente suministraron material e ideas para la elaboración de este artículo.

IX. Referencias

- [1] Alamo J, Dos Santos A., "Desarrollo de un Sistema de Promoción de Empleados basada en Lógica Difusa", Trabajo Especial de Grado para optar al título de Técnico Superior Mención Informática. Instituto Universitario Tecnológico de la Región Capital "Federico Rivera Palacios", 2004.
- [2] Bosc P, Pivert O. *SQLf: A Relational Database Language for Fuzzy Querying*. IEEE Transactions on Fuzzy Systems, Vol 3, (1995) Pp. 1-17.
- [3] Bosc, P., Pivert, O., *SQLf Query Functionality on Top of a Regular Relational Database Management System*, Knowledge Management in Fuzzy Databases, Pons, O., Vila, M. and J. Kacprzyk (Eds.), Physica-Verlag, (2000), Pp. 171-190.
- [4] Camacho F, Quilisque D. "Desarrollo de un Sistema de Diagnóstico de Enfermedades Respiratorias basado en Lógica Difusa", Trabajo Especial de Grado para optar al título de Técnico Superior Mención Informática. Instituto Universitario Tecnológico de la Región Capital "Federico Rivera Palacios", 2006.
- [5] Castillo M, Luna D. "Sistema de Control y Evaluación de Proyectos de Software", Trabajo Especial de Grado para optar al título de Técnico Superior Mención Informática. Instituto Universitario Tecnológico de la Región Capital "Federico Rivera Palacios", 2006.
- [6] Charaima J, Morales D, "Desarrollo de una aplicación web de compra de vehículos basada en lógica difusa", Trabajo Especial de Grado para optar al título de Técnico Superior Mención Informática. Instituto Universitario Tecnológico de la Región Capital "Federico Rivera Palacios", 2004.
- [7] Cox, E., *Relational Database Queries using Fuzzy Logic*, Artificial Intelligent Expert, (1995), Pp. 23-29.
- [8] Fernández A, Ortiz J. "Desarrollo de un Sistema de Evaluación de Desempeño Docente basado en Lógica Difusa", Trabajo Especial de Grado para optar al título de Técnico Superior Mención Informática. Instituto Universitario Tecnológico de la Región Capital "Federico Rivera Palacios", 2006.
- [9] Fukami S, Umamo M, "Fuzzy Relational Algebra for Possibility-Distribution-Fuzzy-Relational Model of Fuzzy Data", Journal of Intelligent Information System, Vol 3, pp 7-27, 1994.
- [10] Furcolo, N., Tineo, L., "SQLfi más flexible y libre", Acta Científica Venezolana, suplemento especial procedente de la LVII Convención Anual de la AsoVAC (2007).
- [11] Galindo J, "New Characteristics in FSQL, a Fuzzy SQL for Fuzzy Databases". WSEAS Transactions on Information Science and Applications 2, Vol. 2, pp. 161-169, February 2005
- [12] Galindo J, Urrutia A, Piattini M, *Fuzzy Databases: Modeling, Design and Implementation*. Book the Idea Group Publishing Hershey, USA. (2006).
- [13] Galindo J. (Ed.), (2008). *Handbook of Research on Fuzzy Information Processing in Databases*. Hershey, PA, USA: Information Science
- [14] Gallón K, Nieto N. "Sistema Automatizado basado en Bases de Datos Difusas para la Elaboración de Evaluaciones Docentes", Trabajo Especial de Grado para optar al título de Técnico Superior Mención Informática. Instituto Universitario Tecnológico de la Región Capital "Federico Rivera Palacios", 2006.
- [15] Goncalves M, Tineo L, "SQLf Flexible Querying Language Extension by means of the norm SQL2", The 10th IEEE International Conference on Fuzzy Systems, Vol 1, Dec 2001.
- [16] Goncalves M, Tineo L, "SQLf3: An extension of SQLf with SQL3 features", The 10th IEEE International Conference on Fuzzy Systems, Vol 3, Dec 2001.
- [17] Goncalves M, Tineo L (2002). *Aplicación de Consultas Difusas para Evaluar Desempeño Docente*. LII Convención anual de la Asociación Venezolana para el Avance de la Ciencia, AsoVAC 2002, pág. 362.
- [18] Goncalves M, Tineo L (2008). *SQLfi y Sus Aplicaciones*. Revista Avances en Sistemas e Informática, Volumen 5, Nro 2.
- [19] Gonzáles, C. "Una Propuesta de Integración y Estandarización de FSQL y SQLf", Trabajo de grado presentado a la Universidad Simón Bolívar Como requisito parcial para optar al grado de Magíster en Ciencias de la Computación, Julio 2008.

- [20] Kacprzyk J, Zadrozny S, "Fuzzy Queries in Microsoft AccessTM v.2", Proc. of Fuzzy IEEE'95 Workshop on Fuzzy Database Systems and Information Retrieval, 1995.
- [21] Lee K, Loo G, "An Interface to Databases for Flexible Query Answering: A Fuzzy-Set Approach". Lecture Notes in Computer Science 1873. DEXA 2000.
- [22] Ma Z.M., Li Yan, *Generalization of Strategies for Fuzzy Query Translation in Classical Relational Databases*. Information and Software Technology, Volume 49, Issue 2, February 2007.
- [23] Nakajima H., Sogoh T., Arao M., "Fuzzy Database Language and Library-Fuzzy Extension to SQL", Proc. of Second IEEE International Conference on Fuzzy Systems, 1983.
- [24] Oceano Langenscheidt, "Gramática Práctica", Oceano Grupo Editorial, Barcelona, España, 1999
- [25] Petry, F., "Fuzzy Databases Principles and Applications", Chapter 6, International Series in Intelligent Technologies, Klumer Academic Publishers, (1996), Pp.184-221.
- [26] Tineo, L., "Interrogaciones Flexibles en Bases de Datos Relacionales", Trabajo de Ascenso para Optar a la Categoría de Profesor Agregado, Universidad Simón Bolívar, Venezuela, (1998).
- [27] Tineo L. "Una Contribución a la Interrogación Flexible de Bases de Datos: Evaluación de Consultas Cuantificadas Difusas", Tesis Doctoral, Universidad Simón Bolívar, Caracas, Venezuela, (2006).
- [28] M. Wong and K. Leung. "A fuzzy Database-Query Language". Information Systems. Vol 15. No. 5, pp 583-590, 1990.
- [29] Zadeh, L. A. "The concept of a linguistic variable and its application to approximate reasoning" Information Science, 8, 1975, pp 199-249.
- [30] Zadeh, L. A. (1977) "PRUF – A Language for the Representation of Meaning in Natural Languages". IJCAI 918.