

Enl@ce: Revista Venezolana de Información,
Tecnología y Conocimiento
ISSN: 1690-7515
Depósito legal pp 200402ZU1624
Año 6: No. 3, Septiembre-Diciembre 2009, pp. 29-46

Cómo citar el artículo (Normas APA):
Rodríguez, R. y Goncalves, M. (2009). Perfil UML para el
modelado visual de requisitos difusos. *Enl@ce:
Revista Venezolana de Información, Tecnología y
Conocimiento*, 6 (3), 29-46

Perfil UML para el modelado visual de requisitos difusos

Rosseline Rodríguez¹

Marlene Goncalves²

Resumen

Modelar dominios de aplicación que involucren requisitos con términos vagos puede ser un proceso complejo, debido a que las herramientas de modelado disponibles en la actualidad no están concebidas para ello. Sin embargo, muchos requisitos de información en sistemas y aplicaciones actuales podrían involucrar términos vagos del lenguaje natural que expresan las preferencias del usuario. Algunos esfuerzos previos se han hecho en el área de consultas a bases de datos incorporando la lógica difusa para la expresión de tales preferencias. En este trabajo, se define un perfil bajo UML que permite el modelado visual de requisitos de información basados en términos lingüísticos, ejemplo de ellos son los llamados requerimientos *fuzzy* o difusos. El perfil propuesto se basa en estereotipos y lógica difusa extendida extendido OCL (*Object Constraint Language*). Dicho perfil posee una semántica formal que permite eliminar ambigüedades, que puede ser representado gráficamente. Se ejemplifica la aplicación del perfil a través de un caso de estudio de la vida real que muestra la sencillez y potencialidad de su uso.

Palabras clave: requisitos difusos, UML, OCL, perfil UML, especificaciones formales

Recibido: 07-10-09 Aceptado: 23-11-09

1 Ingeniera en Computación. MSc en Ciencias de la Computación, Profesor con categoría de Agregado en la Universidad Simón Bolívar en Venezuela. Correo electrónico: crodrig@ldc.usb.ve

2 Licenciada y Doctora en Computación. Profesora Asociado de la Universidad Simón Bolívar en Venezuela. Acreditada al Programa de Promoción al Investigador (PPI) Nivel I. Correo electrónico: mgoncalves@usb.ve

UML Profile for Visual Modeling of Fuzzy Requirements

Abstract

Modeling of application domains that include requirements with vague terms may be a complex process because existing modeling tools are not conceived for that. Nevertheless, many real life applications and systems information requirements might involve natural language vague terms for the expression of user preferences. Some previous works in the field of databases add fuzzy logic for preferences expression in querying. In this work, we define a UML profile that allows visual modeling of information requirements involving vague linguistics terms, i.e., fuzzy requirements. The proposed profile is based on stereotypes and fuzzy logic extended to OCL. This profile has a formal semantic to avoid ambiguity, moreover it provides diagram representation. We illustrate our profile by means of a real-world case study showing the potential and simplicity of this proposal.

Key words: Fuzzy Queries, UML, OCL, UML Profile, Formal Specifications

Introducción

Las necesidades de información de los usuarios pueden expresarse en lenguaje natural mediante frases que involucran términos vagos o imprecisos que representan criterios de preferencia de usuario sobre los datos. En la teoría de conjuntos difusos, los términos vagos o imprecisos corresponden a conceptos difusos cuyos límites no están claramente definidos y/o cuya semántica es susceptible al contexto y/o preferencias del usuario. Estos conceptos pueden ser modelados usando conjuntos difusos (Zadeh, 1965), los cuales fueron introducidos como una representación matemática de conjuntos donde los elementos cercanos al borde no son completamente incluidos ni totalmente excluidos del conjunto. De esta forma, los conjuntos difusos permiten la membresía gradual de los elementos al conjunto.

Para efectos de este documento los requisitos son necesidades de información de los usuarios a ser satisfechos mediante consultas a bases de datos. Estos requisitos podrían ser parte del resultado de un análisis del sistema o aplicación a modelar. No debe confundirse con los requisitos expresados en UML mediante diagramas de casos de uso, pues ellos se refieren a la interacción del usuario con el sistema y no a sus necesidades de información. Cuando los requisitos incluyan términos lingüísticos vagos, se usará la denominación de requisitos difusos.

Para ilustrar la presencia de términos vagos en un requisito difuso, supóngase que alguien está interesado en comprar un vehículo y desea conocer cuáles son los vehículos *económicos*. El adjetivo *económico* es un término vago del requisito. Este término no incrementa la ambigüedad del

requisito ya que la ambigüedad es propia de la ausencia del conocimiento del dominio: los vehículos pertenecen a un concesionario o son autos de una determinada marca. Esta ambigüedad puede ser eliminada si se delimita el dominio del problema. Por ejemplo, cuáles son los vehículos *económicos* de la marca “Ford”. Nótese que la definición del término *económico* depende del criterio del usuario puesto que alguien puede definir un vehículo *económico* como aquel cuyo precio es menor de Bs. 50.000, pero otra persona pudiera pensar que un carro es *económico* si su precio está por debajo de Bs. 100.000.

En los trabajos presentados por Bordogna y Psaila (2008), Bosc y Pivert (1995), Galindo, Urrutia y Piattini (2006), se han extendido el lenguaje de consultas SQL (Structured Query Language) con lógica difusa para la expresión de requisitos que involucran términos lingüísticos vagos o imprecisos. Entre éstas, la propuesta de Bosc y Pivert (1995) es la más completa ya que incorpora la mayor cantidad de elementos de la lógica difusa, así como también es el único que está actualizado a los estándares de SQL 2003 (González, Goncalves y Tineo, 2009). Aunque estos lenguajes han sido incorporados en sistemas manejadores de bases de datos (Eduardo, 2003), existe aún el problema de representar el conocimiento vago e incierto propio de ciertos dominios de aplicación.

El modelo orientado a objetos y el modelo Entidad-Relación han sido dos de los paradigmas más difundidos y utilizados para representar una variedad de situaciones reales. Sin embargo, estos modelos no son capaces de representar dominios de aplicación donde la información vaga e impre-

cisa no puede ser ignorada porque se le considera muy útil para satisfacer ciertos requisitos de usuarios. Por ello, varios trabajos han extendido el modelo orientado a objetos y el modelo entidad relación integrándole características difusas que permitan modelar tales dominios (Bordogna, Lucarella y Pasi, 1994; Cuevas, 2009; George, Buckles y Petry, 1993; Pereira y Tineo, 2000; Urrutia, Varas y Galindo, 2003). Aunque cualquiera de estos modelos permite modelar situaciones del mundo real con presencia de imprecisión o vaguedad, estos modelos no han sido adoptados debido a su complejidad. Por ejemplo, considérese una especialización difusa para modelar una casa rodante; una casa puede heredar características de casa y vehículo con ciertos grados de membresía. El problema de la complejidad está en como definir estos grados de membresía y qué significa que una clase herede gradualmente de otra. En general, para cualquier usuario no es sencillo modelar este tipo de situaciones utilizando estos modelos, pero cualquier usuario si es capaz de expresar sus requisitos difusos usando el lenguaje natural. No obstante, el problema del lenguaje natural es la ambigüedad, la cual puede resolverse a través de la especificación de los requisitos en un lenguaje formal.

En este trabajo se define un Perfil UML para el modelado visual de requisitos difusos. Adicionalmente, para describir la semántica formal de estos requisitos, se hace una extensión con lógica difusa al lenguaje OCL (Object Constraint Language). El uso de este perfil permite eliminar la ambigüedad de dichos requisitos propia del lenguaje natural. El resto de este documento se organiza en cuatro secciones. En la sección 2, se resumen los

fundamentos teóricos. En la sección 3, se define el nuevo perfil UML, así como el OCL extendido con lógica difusa. En la sección 4, se ilustra el uso de este perfil a través de un caso de estudio real. Por último, en la sección 5, se presentan las conclusiones y trabajos futuros.

Fundamentos Teóricos

En esta sección se inicia con los conceptos de la lógica difusa que son la base de la propuesta presentada en este artículo. Posteriormente se describirá detalladamente la definición de requisitos difusos, los términos gramaticales que permiten detectar dichos requisitos dentro de un universo de discurso y las características que debe cumplir una aplicación para que pueda incorporar dichos requisitos en su funcionalidad. Finalmente, se describe cómo definir nuevos perfiles UML.

Lógica difusa

Los conjuntos difusos fueron introducidos por Zadeh (1965). En un conjunto difuso, cada elemento está provisto de un grado que representa su membresía al conjunto. La membresía se define en base a una función cuyo rango es el intervalo real $[0,1]$; la función de membresía de un conjunto difuso F se denota como μ_F .

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”. Esta lógica permite dar una interpretación a términos lingüísticos vagos, a saber:

- *Predicados*, que son los componentes atómicos de esta lógica, 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” (Zadeh, 1975).
- *Modificadores*, tales como los adverbios, la negación y el antónimo, términos que permiten definir predicados modificados por medio de transformaciones sobre funciones de membresía.
- *Comparadores*, una clase de predicados difusos definido sobre pares de elementos, ellos establecen una comparación difusa.
- *Conectores*, operadores definidos para la combinación de condiciones difusas. La negación, conjunción y disyunción difusas son extensiones de sus equivalentes clásicas.
- *Cuantificadores*, términos que describen cantidades, tales como “la mayoría”, “cerca de la mitad”, “alrededor de 20”; éstos son una extensión de los cuantificadores usuales universal y existencial.

Requisitos Difusos

Los requisitos difusos (Rodríguez y Tineo, 2009) involucran términos vagos o imprecisos que representan las preferencias del usuario sobre los datos. Estos términos pueden ser modelados usando conjuntos difusos (Zadeh, 1965).

Un analista de sistemas puede determinar los elementos gramaticales que son indicadores de la vaguedad en los requisitos difusos escritos en lenguaje natural, pues es en este lenguaje donde

se usan los términos vagos. Estos elementos gramaticales pueden definir variables lingüísticas y/o términos difusos, es decir, predicados, comparadores, cuantificadores, modificadores y/o conectores difusos. Por esta razón durante el análisis se determina cuáles términos difusos se usarán, además de su tipo y forma. En la mayoría de los casos, una palabra de naturaleza vaga del lenguaje

natural se modela como un solo término difuso, sin embargo, en algunos casos es necesario usar una expresión más compleja que involucra varios términos difusos.

Los términos lingüísticos vagos a considerar fueron analizados por Rodríguez y Tineo (2009), y su correspondiente modelado se resumen en la **Tabla 1**.

Tabla 1
Términos lingüísticos vagos

Términos	Descripción	Modelado como
Adjetivos calificativos de grado positivo	Añaden una nota de cualidad; el grado positivo corresponde al adjetivo en su forma original, tales como <i>bueno, malo, alto, bajo, económico, costoso, eficaz, productivo, nuevo y antiguo</i> .	Predicados difusos.
Adjetivos calificativos de grado comparativo	Expresan superioridad e inferioridad, tales como <i>más joven, menos productivo, más eficaz, mejor, peor, mayor, menor, superior, inferior, exterior e interior</i> .	Comparadores difusos.
Adjetivos calificativos de grado superlativo relativo	Indican una superioridad o inferioridad relativa a un grupo de entes similares. Se forman con los adjetivos comparativos añadiendo un artículo determinado, así como <i>el más productivo, el menos costoso</i> . También hay algunos puros como: <i>óptimo, pésimo, máximo, mínimo, supremo, sumo, ínfimo, extremo e íntimo</i> .	Cuantificador o comparador difuso.
Adjetivos calificativos de grado superlativo absoluto	Denotan la superioridad o inferioridad ideal, se caracterizan por las terminaciones <i>ísimo y érrimo</i> , tales como <i>buenísimo, malísimo, antiquísimo, altísimo, bajísimo, riquísimo y paupérrimo</i> . Éstos tienen equivalencia con el uso del adverbio muy seguido de un adjetivo calificativo positivo.	Predicado difuso posiblemente afectado por un modificador difuso.
Adjetivos determinativos indefinidos cuantitativos	Añade una nota vaga sobre la cantidad, tales como <i>algunos, bastantes, cuantiosos, demasiados, muchos, múltiples, ningunos, numerosos, pocos, tantos, todos, unos y varios</i> .	Cuantificador difuso.
Adverbios variables	Son palabras que modifican al verbo a un adjetivo u otro adverbio, particularmente nos interesan adverbios como <i>bien, mal, grandemente, pequeñamente, altamente, bajamente, externamente, internamente, óptimamente, pésimamente, máximamente, mínimamente, supremamente, sumamente, ínfimamente y extremamente</i> . Otros ejemplos de este grupo son las palabras <i>cerca y lejos</i> .	Modificador o comparador difuso.
Adverbios invariables de cantidad	Son adverbios que no tienen grado, nos interesan palabras como <i>bastante, harto, medio, muy, poco, mucho, tanto, tan, casi y algo</i> .	Modificador difuso.

Por otro lado, debido a su vaguedad, los requisitos difusos no son fácilmente expresables con condiciones en lógica clásica, por lo que requieren del uso de la lógica difusa como herramienta de modelaje. El analista puede determinar si el siste-

ma es factible de ser modelado utilizando la lógica difusa a través de siete características presentadas por Rodríguez y Tineo (2009), las cuales se resumen en la **Tabla 2**.

Tabla 2
Características de factibilidad para modelar un sistema usando lógica difusa

Característica	Descripción
Inactividad	Cualidad del sistema de ser intuitivo para los usuarios. En los requisitos se observan términos lingüísticos importantes que son muy cercanos al lenguaje natural más que valores, rangos o formulaciones complejas propias de lenguajes de consultas a bases de datos. Éstos deben ser fácilmente entendibles y manejables por el usuario final.
Flexibilidad	Capacidad del sistema 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.
Vaguedad	Engloba la imprecisión o incertidumbre que en general está presente en los sistemas de conocimiento y de razonamiento humano. Se incluye cuando en los sistemas las consultas abarcan proposiciones que 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.
Tolerabilidad	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 se conoce como la calibración (González et als, 2009), la cual limita por un cierto grado la generación de respuestas superpobladas y/o respuestas indeseadas por su bajo nivel de satisfacción o calidad.
Adaptabilidad	El manejo de las preferencias es parametrizable o adaptable por el usuario. Es decir, cada término lingüístico debe ser susceptible a cambios debido a percepciones de los diferentes usuarios o contextos.
Gradualidad	Es importante cuando las respuestas a los requisitos de las aplicaciones son discriminadas por grados, de acuerdo con la satisfacción de las condiciones involucradas. Cada elemento en el conjunto de respuestas resultantes tiene asociado un valor que expresa su importancia o preferencia para el usuario de acuerdo al requisito expresado.
Gerencialidad	Cualidad de un sistema de dar soporte a la toma de decisiones. Aunque los requisitos difusos pueden ser útiles para personas en cualquier nivel de la organización o para usuarios comunes, ellos toman mayor aplicabilidad e importancia en el entorno gerencial, donde los términos difusos aparecen con mayor frecuencia y tienen mayor necesidad de ser definidos. El principio de incompatibilidad de Zadeh (1975) lo describe así "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.

Perfiles UML

UML (Unified Modeling Language), definido por Booch, Rumbaugh y Jacobson (2005), es un lenguaje de propósito general usado para especificar, visualizar y construir artefactos de sistemas de software. Un artefacto se corresponde a un modelo o pieza de información producido en el proceso de desarrollo de software.

Dado que UML es un lenguaje de propósito general, la descripción de dominios de aplicación específicos requiere de la definición de un nuevo lenguaje que describe el metamodelo, utilizando el estándar MOF (OMG, 2006b), o la extensión propiamente de UML a través de un mecanismo denominado Perfiles UML. Los perfiles UML proveen un mecanismo de extensión genérico para construir modelos UML en dominios particulares. Están basados en estereotipos, restricciones y valores etiquetados adicionales que son aplicados a los elementos o relaciones de un diagrama. En primer lugar, un estereotipo es un nuevo elemento del lenguaje extendido que se define sobre algún elemento pre-existente de UML. Gráficamente, los estereotipos se denotan entre comillas francesas («stereotype») dentro de cajas. En segundo lugar, los estereotipos pueden tener restricciones que imponen condiciones sobre los elementos estereotipados. Para la especificación de restricciones puede utilizarse el lenguaje natural o un lenguaje formal como OCL (Object Constraint Language) propuesto por OMG (2006a). En tercer lugar, los valores etiquetados son meta-atributos asociados a una metaclass extendida por el perfil. Un valor etiquetado se caracteriza por un nombre y un tipo, y se representa como un atributo de la clase que define el estereotipo.

Aunque la definición de un nuevo lenguaje permite una mayor expresividad para modelar los conceptos de dominios particulares, se adoptará la opción de perfiles como solución. La solución basada en perfiles tiene la ventaja de que las herramientas de UML existentes manejan de forma natural los conceptos del dominio específico puesto que la extensión de UML está basada en el metamodelo estándar de UML. Por lo tanto, se propone extender UML con un perfil que permite representar y visualizar los requisitos difusos. El perfil propuesto se basa en estereotipos y en el uso del lenguaje OCL para la especificación formal de tales requisitos.

Lenguaje OCL

OCL (Object Constraint Language) es un lenguaje formal, propuesto por OMG (2006a), usado para describir expresiones sobre UML. Típicamente, estas expresiones modelan condiciones invariantes que el sistema debe cumplir, pero también pueden ser usadas para modelar pre y post condiciones, así como también, consultas sobre los objetos del modelo. Las restricciones en OCL pueden ser colocadas o no dentro del modelo gráfico. Cuando el modelo no es suficientemente expresivo y/o se quiere evitar estados indeseables del sistema, OCL permite la especificación formal de invariantes de los objetos. Los invariantes son expresiones lógicas cuyo resultado debe ser verdadero para todas las instancias de un tipo.

Las expresiones OCL son de la forma **context** *TypeName* **inv** *Expression*, siendo **context** e **inv** palabras reservadas del lenguaje; *TypeName* el nombre de la clase que representa

el contexto y *Expression* la restricción cuyo resultado es un valor booleano. Si la restricción es colocada en el diagrama (dentro del estereotipo apropiado) y las líneas punteadas están conectadas a su correspondiente elemento contextual, no es necesario colocar la declaración del contexto dentro de la restricción. Es decir, la declaración del contexto es opcional. La palabra reservada “self” indica un objeto dentro del contexto actual. Cuando el contexto es claro, esta palabra puede ser eliminada. Una alternativa a “self” es declarar una variable en el contexto para que juegue el rol de “self”. Así, la expresión OCL sería **context** *var*: *TypeName* **inv** *Expression*, donde *var* es el nombre de la variable que sustituye a “self”. Se utiliza la palabra reservada “**pre**” precediendo la expresión, cuando se desea señalar que es una precondition, mientras que para el caso de una postcondición, la palabra reservada es “**post**”. Para indicar que la expresión es el resultado de una consulta se utiliza la palabra reservada “**body**”.

OCL permite el uso de tipos básicos, colecciones y modelos. Entre los tipos básicos se tienen *Integer*, *Real*, *String* y *Boolean*. Los tipos colección, tales como *Set*, *OrderedSet*, *Bag*, *Sequence* y *Collection*, se corresponden a tipos abstractos que contienen operaciones comunes: *size*, *count*, *includes*, *includesAll*, *isEmpty*, *notEmpty*, *sum*, *exists*, *forAll*, *including*, *excluding*, entre otras. Para indicar que una operación se aplica a una colección se utiliza el símbolo “->”. Finalmente, los tipos modelos son las clases, interfaces y tipos definidos en UML.

Propuesta de modelaje visual de requisitos difusos

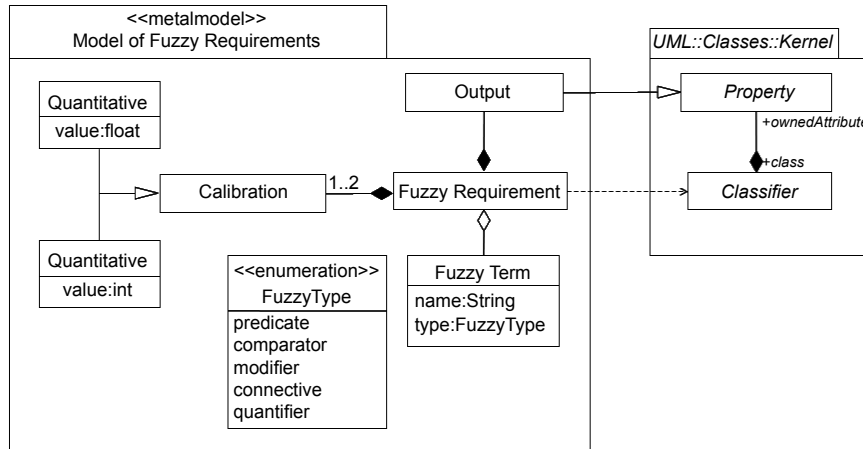
En este trabajo se hace uso de las herramientas de UML para el modelado de datos, describiendo un nuevo perfil que permite representar requisitos que involucran términos vagos o imprecisos, los cuales representan las preferencias del usuario. En la definición del perfil se sigue la metodología propuesta por Giachetti, Marin, y Pastor (2008).

Perfil

El metamodelo que describe el dominio de aplicación puede ser especificado en UML, como se muestra en la **Figura 1**. El tipo de modelo será indicado usando el estereotipo “Model of Fuzzy Requirements” o a través de una nota de UML.

Un requisito difuso (“Fuzzy Requirement”) está formado por un conjunto de términos difusos (“Fuzzy Term”) o vagos, que representan las preferencias del usuario sobre los datos. Los términos difusos poseen un nombre y su tipo puede ser: predicado (“predicate”), modificador (“modifier”), comparador (“comparator”), conector (“connective”) o cuantificador (“quantifier”). En un requisito difuso se especifica además el valor de la calibración cuantitativa (“Quantitative”) y/o cualitativa (“Qualitative”). Adicionalmente, un requisito difuso contiene un conjunto de atributos seleccionados (“Output”) correspondientes a los atributos que el usuario desea que se muestre como respuesta al requisito.

Figura 1
Metamodelo del modelo de requisitos difusos



Un requisito está vinculado con algunas clases y/o asociaciones del modelo de datos. Por lo tanto, en el metamodelo se aprecia que existe una asociación entre la metaclassa “Fuzzy Requirement”, y la metaclassa “Classifier” que generaliza clases y asociaciones en el metamodelo UML, según OMG (2007).

A partir de este metamodelo, el Perfil UML propuesto se describe como un paquete UML denominado “Profile of Fuzzy Requirement”, estereotipado «Profile», tal como se muestra en la **Figura 2**. En el Perfil se especifica cuáles elementos del metamodelo de UML pueden ser asociados con los estereotipos, mediante el uso de flechas continuas. En dicho Perfil de UML, se definen

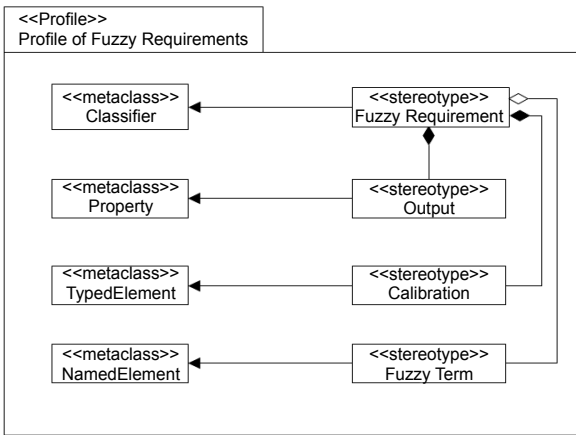
cuatro estereotipos: “Fuzzy Requirement”, “Fuzzy Term”, “Calibration” y “Output”.

El estereotipo “Fuzzy Requirement” extiende la metaclassa “Classifier”. El nuevo estereotipo representa las instancias resultantes de aplicar el requisito. Además se definen otros estereotipos (indicados entre comillas) para representar los diferentes componentes de un requisito difuso. Cada instancia de “Fuzzy Requirement” tendrá el grado de satisfacción que cumple la calibración (“Calibration”) especificada. La calibración puede ser cuantitativa (“Quantitative”) o cualitativa (“Qualitative”), y su valor se especifica a través de un valor etiquetado (“value”). Ésta extiende la metaclassa “TypedElement” que representa los

elementos cuyos valores están restringidos a ser instancias de un tipo. Por otro lado, el estereotipo “Output” extiende los atributos, representados en el metamodelo UML con la metaclassa “Property”. Dichos atributos son los escogidos para participar en la respuesta al requisito.

Un término difuso (“Fuzzy Term”) tiene dos valores etiquetados: el nombre (“name”) del término y el tipo (“type”) del mismo. Este estereotipo es una extensión de la metaclassa “NamedElement” que representa los elementos que pueden tener identificador. Finalmente, se cuenta con un nuevo tipo enumerado “FuzzyType” cuyos valores son el conjunto {predicate, comparator, modifier, connective, quantifier} para indicar el tipo (“type”) del término difuso. Los términos difusos aparecen en las condiciones de búsqueda de los requisitos, las cuales son expresadas mediante el uso de lógica difusa.

Figura 2
Perfil de requisitos difusos



Restricciones

Las restricciones propias del metamodelo, se especifican con el lenguaje OCL de la siguiente manera:

- Todo requisito difuso incluye al menos un término difuso, es decir, el conjunto de términos difuso no puede ser vacío;

context FuzzyRequirement
inv: self.FuzzyTerm->notEmpty()

- Si no se especifican atributos en el resultado se asume que aparecen en la respuesta todos los atributos de las clases y asociaciones de las que depende el requisito;

context FuzzyRequirement
post: self.Output@pre->IsEmpty() **implies**
self.Output = union(self.Classifier->owned
Attribute)

- El valor de una calibración cualitativa está estrictamente contenido entre 0 y 1;

context Qualitative
inv: self.value > 0 **and** self.value < 1

- El valor de una calibración cuantitativa es mayor o igual a 1;

context Quantitative
inv: self.value ≥ 1

- Los atributos que aparecen en el resultado sólo pueden ser de las clases y/o asociaciones de las que depende el requisito.

context FuzzyRequirement
inv: self.Output->forall(a | self.Classifier->
exist(c | a.class = c))

El nuevo perfil agrega restricciones semánticas al metamodelo UML, las cuales son derivadas de las restricciones del metamodelo de Requisitos Difusos. Estas restricciones garantizan que se construyan bajo este perfil, modelos “bien formados”. Dichas restricciones se escriben en el lenguaje OCL como sigue:

context UML::InfrastructureLibrary::Core::Constructs::Classifier

inv: -- el conjunto de términos difusos de un requisito difuso es no vacío

self.isStereotyped(“Fuzzy Requirement”)

implies

self.connection->select(participant.isStereotyped(“Fuzzy Term”))->notEmpty()

post: -- en el resultado aparecen todos los atributos cuando inicialmente se especifica vacío

(self.isStereotyped(“Fuzzy Requirement”) **and** self.connection->select(participant.isStereotyped(“Output”))@pre->IsEmpty())

implies

self.connection->select(participant.isStereotyped(“Output”))

= **union**(self.connection->select(isStereotyped(“Dependency”))->participant.ownedAttribute)

inv: -- el valor de una calibración cualitativa está estrictamente contenido entre 0 y 1

self.isStereotyped(“Fuzzy Requirement”)

implies

(self.connection->select(participant.isStereotyped(“Qualitative”)).value > 0 **and**

self.connection->select(participant.

isStereotyped(“Qualitative”)).value < 1)

inv: -- el valor de una calibración cuantitativa es mayor o igual a 1

self.isStereotyped(“Fuzzy Requirement”)

implies

self.connection->select(participant.isStereotyped(“Quantitative”)).value ≥ 1

inv: -- los atributos del resultado sólo pueden ser de clases de las que depende el requisito. self.

isStereotyped(“Fuzzy Requirement”) **implies**

self.connection->select(participant.isStereotyped(“Output”))->

forall(a | self.connection.isStereotyped(“Dependency”))->**exist**(c | a.class = c))

inv: -- un requisito difuso existe si hay al menos un “Classifier” del cual depende

self.isStereotyped(“Fuzzy Requirement”)

implies

self.connection.oclIsKindOf(Dependency)->notEmpty()

Los requisitos difusos pueden ser representados gráficamente como cajas según el perfil UML de modelado de datos (Ambler, 2009; OMG, 2007). Bajo este perfil UML de modelado de datos, los requisitos difusos deben estar vinculados a clases (correspondiente a las metaclasses del kernel UML “Class” o “AssociationClass”), sobre las cuales se realiza el requisito. Dado que la asociación entre un requisito difuso y alguna de estas clases es una relación de dependencia, ésta se representa gráficamente con líneas punteadas que terminan en una flecha abierta.

La **Figura 3** muestra los componentes de la representación gráfica de un requisito difuso.

En la caja aparece el identificador del requisito, los meta-atributos (fuzzy terms y calibration) que son únicos para la clase, los atributos presentes en cada instancia del requisito y la expresión OCL que indica la semántica del requisito.

Figura 3
 Componentes del requisito difuso presentes en su representación gráfica

<<Fuzzy Requirement>> Nombre
Meta-atributos ...
Atributos ...
Semántica en OCL ...

La semántica del requisito puede ser definida en lenguaje natural o a través de una expresión en el lenguaje OCL (OMG, 2006a), tipo **body**. La operación **select** de OCL permite especificar las instancias que formarán parte de la respuesta del requisito de acuerdo a una expresión en lógica clásica. Dado que se desean incluir términos vagos en estas expresiones se debe extender el lenguaje para que permita expresiones con lógica difusa. Este lenguaje, que ha sido usado hasta ahora para describir formalmente las restricciones dentro de los elementos del modelo de datos, gracias al aporte del presente trabajo servirá además para especi-

ficar la semántica formal de requisitos difusos. De esta forma se aumenta la expresividad del modelo de datos pues se incluye los requisitos, a la vez que se elimina la ambigüedad de tales requisitos, pues se realiza la especificación de los mismos usando OCL.

Las expresiones OCL de los requisitos difusos incluirán los términos vagos que fueron declarados con el estereotipo <<Fuzzy Term>> o con alguno de los literales del tipo enumeración “FuzzyType”. Las nuevas condiciones válidas que forman parte de las expresiones lógicas son de la forma $att\ is\ fp$, $att\ is\ fm\ fp$, $att_1\ fc\ att_2$, donde att, att_1, att_2 son atributos válidos que pertenecen a las clases de las cuales depende el requisito, fp es un predicado difuso declarado, fm un modificador difuso declarado y fc un comparador difuso declarado. Los conectores difusos aparecerían en el mismo lugar donde se utilizan los conectores clásicos como **and** y **or**, es decir, $exp_1\ fc\ exp_2$, siendo exp_1 y exp_2 dos expresiones válidas en lógica difusa y fc un conector difuso declarado. Note que cuando los conectores clásicos tienen como operandos expresiones en lógica difusa, su semántica pasa a ser difusa, es decir el mínimo para el **and** y el máximo para el **or**. En el caso de los cuantificadores difusos, éstos corresponden a operaciones sobre colecciones, cuya sintaxis es similar a la de los cuantificadores clásicos **forall** y **exist** de OCL, es decir, $collection\ ->fq(v\ |\ logic-expression-with-v)$ donde fq es un cuantificador declarado en el requisito difuso y la expresión interna en lógica clásica o difusa, se aplica a cada uno de los componentes de la colección, representados por la variable v .

Sistema de compra de autos en línea

A fin de ejemplificar el uso del perfil propuesto, se planteó el desarrollo de una aplicación Web para la compra y venta de vehículos en línea, de acuerdo a las preferencias y/o necesidades del usuario. El análisis del universo de discurso produce una lista de requisitos difusos en lenguaje natural, con los términos difusos identificados. Posteriormente se determina si el sistema cumple con las siete características que indican si éste es factible de ser modelado con lógica difusa. Luego, los términos lingüísticos de naturaleza vaga se representan usando la teoría de la lógica difusa, indicando el tipo al que pertenecen. Cada requisito difuso se modela según el nuevo perfil “Fuzzy Requirement”, de esta forma serán incluidos en el modelo de clases del sistema. Finalmente, se modela la semántica del requisito usando el lenguaje OCL.

Análisis

El sistema debe permitir el registro de usuario; la oferta de vehículos a través de anuncios con el detalle de sus características y accesorios; y la búsqueda de vehículos con los criterios de preferencia definidos por el usuario que luego serán añadidos a una lista de vehículos preferidos para ser consultados posteriormente.

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 sería una lista de los vehículos encontrados ordenada en forma descendente de acuerdo a la satisfacción del requisito difuso. Algunos requisitos interesantes que se quieren realizar al sistema son:

- Cuáles son los vehículos *nuevos*
- Cuáles son los vehículos potencia *alta*
- Cuáles son los vehículos de precio *bajo*
- Cuáles son los vehículos *pequeños*, de precio *bajo* y potencia *alta*
- Cuáles son las marcas con la *mayoría* de los vehículos de color *claro*
- Cuáles son los modelos de mis marcas *favoritas*, tales que la *mayoría* de los vehículos tienen precio *muy bajo* y color *similar a gris*

Características difusas presentes en esta aplicación

Como parte del análisis se debe verificar que la aplicación es factible de ser modelada con lógica difusa a través de las siete características definidas por Rodríguez y Tineo (2009). Esto se realiza observando los requisitos en el universo de discurso. Para el presente caso de estudio se satisfacen de la siguiente manera:

- *Intuitividad*: Se requiere de una interfaz visual e intuitiva semejante a los sistemas clásicos de compra-venta de vehículos, la cual facilite la especificación de preferencias para datos como la potencia, el modelo, la marca, el color, sin necesidad de conocer la teoría de lógica difusa. El sistema se espera que responda preguntas

intuitivas como ¿cuáles son los vehículos más económicos?

- **Flexibilidad:** Se desea que en los criterios de búsqueda puedan especificarse tanto valores precisos como difusos. En las respuestas se espera que se incluyan elementos en el borde del conjunto aunque no se ajuste perfectamente al ideal. El sistema debe ser flexible.
- **Vaguedad:** En los requisitos se incluyen términos vagos como “nuevo”, “alto”, “bajo”, “pequeño”, “muy”, “la mayoría”, “similar a” para la diversa información manejada. La presencia de tales términos muestra la relevancia de la vaguedad en esta aplicación.
- **Tolerabilidad:** Debido a la gran cantidad de información que puede ser manejada en un sistema de este tipo, es deseable que el usuario pueda especificar la precisión de la búsqueda (calibración cualitativa) y cantidad de resultados (calibración cuantitativa) en los diferentes requisitos que plantee. Esto garantizaría la calidad de las respuestas.
- **Adaptabilidad:** Es conveniente que la aplicación provea mecanismos para criterios de búsqueda predefinidos y modificables según las preferencias del usuario. De esta forma es adaptable a cada contexto e interpretación particular.
- **Gradualidad:** Los resultados de las consultas deberían ordenarse de acuerdo al grado de satisfacción establecido por el usuario según sus preferencias, esto le facilitaría la toma de decisiones.
- **Gerencialidad:** Aunque el usuario en principio no se espera que participe en la toma de deci-

siones gerenciales, este sistema puede ser usado para analizar las ventas a nivel nacional, de ciudades, concesionarios, entre otros.

Términos vagos

Los términos **nuevo**, **alto**, **bajo**, **pequeño**, **claro** y **favorito** son sin duda los más sencillos. Éstos son adjetivos calificativos de grado positivo, por lo que se modelan como predicados difusos. El término **similar** es un adjetivo calificativo de grado comparativo, lo que nos indica un comparador difuso. El término **muy** es adverbio invariable de cantidad, que claramente aparece modificando el adjetivo **bajo**, por lo que es un modificador difuso. La expresión lingüística **la mayoría** es un adjetivo determinativo indefinido cuantitativo, por lo que corresponde a un cuantificador difuso, además, es proporcional pues describe una cantidad relativa a un todo.

Cada uno de estos términos pertenece al estereotipo <<Fuzzy Term>> y se declaran en el requisito según su tipo identificado.

Modelo de dominio extendido

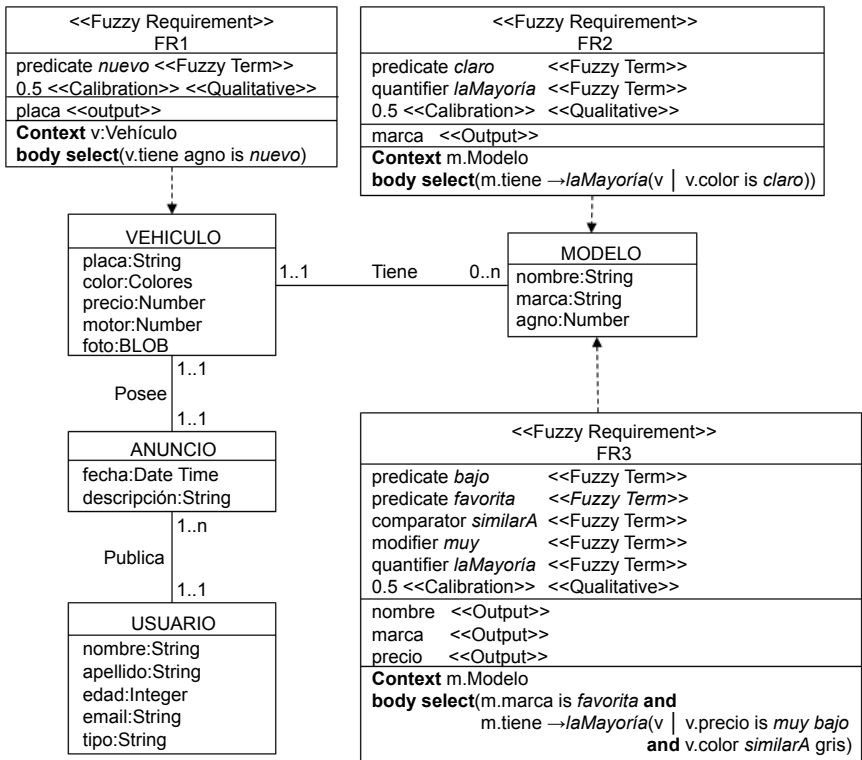
En el modelo de dominio del sistema, **Figura 3**, se identifican cuatro clases principales que permiten resolver los requisitos del sistema: el **USUARIO** que contiene sus datos personales y un tipo que indica si es un cliente o un vendedor; el **VEHÍCULO** con los detalles individuales y técnicos de los vehículos ofertados; el **MODELO** que almacena la información de los modelos de las diferentes marcas presentes en la base de datos; y el **ANUNCIO** que guarda la información a ser publicada sobre los vehículos oferta-

dos. Además se ha incluido en este modelo tres de los requisitos difusos, usando los estereotipos del perfil “Profile of Fuzzy Requirements”: FR1 (Cuáles son los vehículos *nuevos*), FR2 (Cuáles son las marcas con la *mayoría* de los vehículos de color *claro*) y FR3 (Cuáles son los modelos de

mis marcas *favoritas*, tales que la *mayoría* de los vehículos tienen precio *muy bajo* y color *similar a gris*).

Para el primer requisito se ha declarado el término difuso “nuevo” como un “predica-
te”, tal como se identificó en la sección anterior.

Figura 4
Modelo de dominio extendido



Éste depende de la clase Vehículo y solo se coloca como resultado su atributo “placa”. La semántica del requisito es colocada como una expresión OCL de tipo **body**. El segundo requisito depende de las clases Vehículo y Modelo, escogiéndose el atributo “marca” de esta última para aparecer en el resultado. Aquí, adicionalmente se tiene la declaración de un cuantificador difuso “*laMayoría*”. Finalmente, en el tercer requisito se declaran varios tipos de términos difusos, así como una combinación de atributos de las clases Vehículo y “Modelo” de las cuales depende el requisito. En los tres requisitos se especifica la calibración 0.5 de tipo cualitativa.

En la **Figura 4** se han incluido todos los elementos del nuevo perfil a manera de ejemplo, lo cual puede hacer parecer el modelo un poco recargado. También puede entenderse que es necesario conocer la semántica formal del requisito desde el comienzo. Sin embargo, estos elementos pueden irse agregando progresivamente a medida que el diseñador va conociendo el universo de discurso de la misma manera como se realiza con los diferentes modelos UML. Además, no es obligatorio colocar todos los elementos. Puede preferirse extraer la expresión OCL mediante una nota anexa UML, como se hace en otros modelos que utilizan expresiones OCL.

Conclusiones

El principal aporte de este trabajo constituye la definición formal de un perfil UML que permite el modelado de requisitos difusos sobre un

modelo orientado a objetos. Estos requisitos involucran términos lingüísticos vagos que expresan las preferencias del usuario. El nuevo perfil puede ser incluido en el modelado visual del sistema y tiene una semántica formal que permite eliminar la ambigüedad presente en las descripciones en lenguaje natural de los requisitos de información. Este perfil fue definido a través de sus estereotipos, restricciones y valores etiquetados, mecanismo provisto por UML para las extensiones a su metamodelo.

Otro aporte de esta investigación es la extensión del lenguaje OCL para la especificación formal de la semántica de requisitos difusos. Para esto fue necesaria la incorporación de elementos de la lógica difusa en OCL a fin de incluir los términos vagos presentes en dichos requisitos. En este trabajo sólo se presenta la declaración de los términos difusos y no su definición, ya que ésta depende de las preferencias del usuario y puede cambiar al momento de la ejecución. Queda abierto el camino de la definición formal de cada uno de estos términos considerando las preferencias de usuario, así como su incorporación dentro del modelo de datos. Se aplicó este perfil en el modelado de un caso de estudio de la vida real: un sistema para la compra de autos en línea, lo cual mostró su sencillez y potencialidad de uso.

Finalmente, como trabajos futuros se plantean definir una metodología para el desarrollo de aplicaciones con requisitos difusos basada en el diseño de modelos de datos, en la cual se incorpore los resultados de trabajos previos sobre la imple-

mentación de consultas difusas a bases de datos relacionales.

Bibliografía

- Ambler, S. W. (2009). A UML Profile for Data Modeling. Agil Data. Ambysof. Recuperado el 10 de octubre de 2009, del sitio Web: <http://www.agiledata.org/essays/umlDataModelingProfile.html>
- Booch, G., Rumbaugh, J. y Jacobson, I. (2005). *The Unified Modeling Language User Guide*. Addison Wesley, Reading; MA
- Bordogna, G., Lucarella, D. y Pasi, G. (1994). A Fuzzy Object Oriented Data Model. *Proceedings of IEEE 3rd International Conference of Fuzzy Systems*. Vol 1, (1), 313-318
- Bordogna, G. y Psaila, G. (2008). *Customizable Flexible Querying for Classical Relation Databases*. Pensilvania, USA: Galindo, J (Ed.) Handbook of Research on Fuzzy Information Processing in Databases. Vol 1, (1), 191-217
- Bosc, P. y Pivert, O. (1995). SQLf: A Relational Database Language for Fuzzy Querying. *IEEE Transactions on Fuzzy Systems*. Vol 3, (1), 1-17
- Cuevas, L. (2009). *Modelo Difuso de Bases de Datos Objeto-Relacional: Propuesta de Implementación en Software Libre*. Tesis Doctoral presentada ante la Universidad de Granada. Granada, España
- Eduardo, J. (2003). *Sistema de Interrogación Flexible a Bases de Datos SQLf2 – SQLf3*. Informe Final de Trabajo Especial de Grado presentado ante la Universidad Simón Bolívar. Caracas, Venezuela
- Galindo, J., Urrutia, A. y Piattini, M. (2006). *Fuzzy Database Modeling, Design and Implementation*, Idea Group Publishing
- George, R., Buckles, B. y Petry, F. (1993). Modeling class hierarchies in the fuzzy object-oriented data model. *International Journal. Fuzzy Sets and Systems*. Vol. 60. (1), 259-272
- Giachetti, G., Marin, B. y Pastor, O. (2008). Perfiles UML y Desarrollo Dirigido por Modelos: Desafíos y Soluciones para Utilizar UML como Lenguaje de Modelado Específico de Dominio. *Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos*. Vol. 2, (3)
- González, C, Goncalves, M. y Tineo, L. (2009). A New Upgrade to SQLf: Towards a Standard in Fuzzy Databases. *Proceedings of XX International Workshop on Database and Expert Systems Applications*. Vol. 1, (1), 442-446
- Object Management Group (2006a). *Meta Object Facility (MOF) Core Specification OMG*. Recuperado el 9 de octubre de 2009, del sitio Web del Object Management Group: <http://www.omg.org/spec/MOF/2.0/>
- Object Management Group (2006b). *Object Constraint Language Specification, version 2.0*. Recuperado el 9 de octubre de 2009, del sitio Web del Object Management Group: <http://www.omg.org/technology/documents/formal/ocl.htm>
- Object Management Group (2007). *OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2*. Recuperado el 15 de octubre de 2009, del sitio Web del Object Management Group: <http://www.omg.org/spec/UML/2.1.2/>
- Pereira, W. y Tineo, L. (2000). Modelo Orientado a Objetos Difuso. *Acta Científica Venezolana*. Vol. 51. (2), 357
- Rodríguez, R. y Tineo, L. (2009). Elementos Gramaticales y Características que Determinan Aplicaciones con Requerimientos Difusos. *Revista Tekhne*, Vol. 12, (1), Universidad Católica Andrés Bello. Caracas, Venezuela

Urrutia, A., Varas, M. y Galindo, J. (2003). Diseño de una base de datos difusa modelada con UML. *Proceedings of 6° Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes Software*. Vol. 1, (1), 145-155

Zadeh, L. A. (1965). Fuzzy Sets. *Information and Control*. Vol. 8, (3), 338-353

Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning. *Information Science*. Vol. 8, (1), 199-249

Agradecimiento

Es necesario dar reconocimiento al Fondo Nacional de Ciencia, Tecnología e Investigación, FONACIT que ha apoyado esta investigación a través de la subvención científica número G-2005000278 correspondiente al proyecto titulado “Creación y Aplicación de Manejadores de Bases de Datos Difusas”.