Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Implementación de extensiones simbólicas al lenguaje SQL en servidores de bases de datos objeto-relacionales

Implementation of SQL Symbolic Extensions on Object-Relational Database

Johnny Villalobos-Murillo
jvillalobos@una.cr
Escuela de Informática, Universidad Nacional.
Heredia, Costa Rica.

Steven Brenes-Chavarría
sbrenes@una.cr
Escuela de Informática, Universidad Nacional.
Heredia, Costa Rica.

Recibido-Received: ***28/may/2014*** / Aceptado-Accepted: ***21/ago/2014*** / Publicado-Published: ***31/jul /2015.***

Resumen

Este artículo propone extender el lenguaje SQL al crear nuevos tipos de datos con sus respectivos operadores, que permitan crear y manipular objetos simbólicos directamente sobre las bases de datos. La funcionalidad de la extensión propuesta se valida realizando transformaciones de datos relacionales clásicos a objetos simbólicos en bases de datos de gran volumen. Para los usuarios no familiarizados con el lenguaje SQL, se construye una interfaz de usuario final, que facilita y guía el proceso de transformación simbólica.

Palabras claves: Base datos; extensión; objeto-relacional; objeto simbólico; SQL.

Abstract

This article proposes extend the SQL language with new data types and operators for create and manipulate symbolic objects directly in the database. The functionality of the proposed extension is validated by performing transformations of classical symbolic objects to relational data bases of large data. For users not familiar with the SQL language, a final user interface is built to facilitate and guide the process of symbolic transformation.

Keywords: Database; extension; relational-object; SQL; symbolic-object.

Actualmente, los grandes volúmenes de información almacenada presentan dificultades para su manipulación y para la extracción del conocimiento contenido en las bases de datos. Este motivo ha llevado a muchos investigadores de diversos campos, entre los que destacan la estadística y la computación, a enfocar sus esfuerzos para simplificar las tareas de reducción, extracción y análisis de datos. Un ejemplo de este tipo de investigaciones es el proyecto SODAS (Bock H-H & Diday E, 2000) donde se propone la creación y utilización de un nuevo tipo de dato llamado objeto simbólico, el cual se obtiene a partir de bases de datos relacionales (Codd, 1970).

El objeto simbólico puede considerarse como un mecanismo que permita la extracción y agrupación de datos contenidos en bases de datos relacionales. A partir de él, se han modificado algunos métodos de análisis de datos clásicos y se han desarrollado nuevos métodos de análisis simbólico. Tal es el caso de la modificación realizada al algoritmo de análisis de componentes principales para aplicarse a objetos simbólicos (Rodríguez, 2000). Además, se han creado herramientas computacionales para transformar datos clásicos en datos simbólicos; la herramienta *Data Bases to Symbolic Object* (DB2SO, por sus siglas en inglés) del proyecto SODAS es una de ellas.

Los objetos simbólicos producidos por estas herramientas se usan exclusivamente como insumos de los métodos de análisis simbólico, esto hace que no sea posible manipularlos ni utilizarlos en otras áreas de investigación. Con el propósito de solventar esta limitación, nuestra investigación se propuso desarrollar una extensión para el lenguaje SQL. La finalidad es crear y manipular objetos simbólicos con la misma facilidad que presentan los tipos de datos propios de una base de datos, para así extender el uso de estos objetos simbólicos a múltiples áreas de investigación.

Esta investigación propone el diseño e implantación de una extensión para el lenguaje SQL que utilice algoritmos de transformación de datos clásicos a datos simbólicos, el diseño de las estructuras de datos y los operadores necesarios para su persistencia y manipulación en sistemas gestores de bases de datos. Por otro lado y pensando en aquellos usuarios que no estén familiarizados con el lenguaje SQL, se desarrolla una interfaz gráfica que utiliza la extensión propuesta para facilitar el proceso de transformación. La extensión y la interfaz se ponen a disposición como software libre en el sitio WEB del Laboratorio de Bases de Datos de la Universidad Nacional de Costa Rica (www.slinfo.una.ac.cr), contribuyendo, de esta forma, con el libre acceso a las tecnologías de información.

Este artículo está compuesto de cuatro apartados. El primero de ellos estudia formalmente el concepto del objeto simbólico, así como las definiciones matemáticas más importantes. El segundo presenta la extensión simbólica propuesta, lo que incluye los componentes, funciones y sintaxis definida para su utilización en bases de datos. En el tercer apartado se realizan las pruebas de transformación para convertir datos clásicos a datos simbólicos. En el cuarto y último apartado se exponen las conclusiones de esta investigación.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Objetos simbólicos

En esta sección se presentan las definiciones formales y ejemplos de los objetos simbólicos. Los objetos simbólicos pueden ser obtenidos de diversas fuentes, una de las más frecuentes es las tablas de bases de datos relacionales. En la tabla 1, se muestra un ejemplo de datos clásicos que corresponde a un conjunto de observaciones Ω, donde cada tupla de la tabla corresponde a un individuo particular.

Tabla 1
Representación de datos clásicos

A	B	C	D	E
1	34	A	58	68
1	34	A	56	64
2	34	A	54	62
2	36	A	57	65
3	34	B	52	69
3	31	B	55	62
4	36	B	57	67

Fuente: propia del estudio.

Definición 1: Un concepto *C* es el resultado de una intensión *I* sobre una extensión *E*. La intensión es el deseo de encontrar conjuntos de datos a partir de las propiedades de las variables que interesan ser agrupadas y analizadas. La extensión, por su parte, es el conjunto de todos los individuos que cumplan con la intensión.

Suponga el concepto *C* [A, C] con I (A > 1.5, 33≤ B ≤ 36), el resultado es un subconjunto del producto cartesiano $[A \times B]$, que cumple con la intensión *I* y su correspondiente extensión *E*. Se genera la tabla 2 como resultado de aplicar sobre la tabla 1 el concepto *C*.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Tabla 2
Representación de un concepto

A	B
2	34
2	36
3	34
3	36
4	34
4	36

Fuente: propia del estudio.

Una definición no formal establece que los objetos simbólicos se obtienen como el resultado de realizar agrupaciones de datos clásicos en variables simbólicas según un concepto.

Objeto-simbólico → [Concepto] {variables simbólicas,….}

Definición 2: Un vector descriptor d es el conjunto de variables de cada individuo i de la tabla de observaciones Ω. La tabla de observaciones representa el origen de datos clásicos, tome como ejemplo la tabla 1 de la cual se pueden extraer los vectores descriptores $d1 = (1{,}34, A, 58{,}68)\ y\ d5 = (3{,}34, B, 52{,}69)$, para los respectivos individuos $i = 1$ e $i = 5$ en Ω.

Definición 3: Una relación R es una asociación de un vector descriptor d con una variable Y, se puede escribir de la forma YRd. La relación R toma la forma de $=, \leq, <, \geq, >, \neq, \in, \notin$.

Definición 4: Un objeto simbólico es la tripleta $s = (a, R, d)$, donde a es una función de aserción a: $\Omega \rightarrow L$, dicha función mapea cada vector descriptor d del individuo ***i*** del conjunto de observaciones Ω al conjunto L $\{0,1\}$.

Definición 5: Un objeto simbólico se clasifica en binario si $L = \{0, 1\}$, es decir, ***a*** es un mapeo binario, entonces, ***s*** es un objeto simbólico binario.

Definición 6: Un objeto simbólico se clasifica en modal si $L = [0, 1]$, es decir, ***a*** es un mapeo modal, entonces, ***s*** es un objeto simbólico modal.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Clasificación de los objetos simbólicos

Basándose en la cardinalidad del conjunto de las variables que definen su concepto, los objetos simbólicos pueden ser clasificados como objetos de primer orden, segundo orden u orden superior (Diday, 2009).

Definición 7: Los objetos simbólicos son de ***primer orden*** cuando el concepto se refiere a individuos (una sola variable). Un objeto simbólico es de ***segundo orden o superior*** cuando el concepto está formado por clases (dos o más variables).

En la tabla 3 se muestra un objeto simbólico de primer orden $[A]$ tomando como referencia la tabla 1. En la tabla 4 se observa un objeto simbólico de orden superior $[A, C]$.

Tabla 3
Objeto simbólico de primer orden $[A]$

Concepto
$i_1(1)$
$i_2(2)$
$i_3(3)$
$i_4(4)$

Fuente: propia del estudio.

Tabla 4
Objeto simbólico de orden superior $[A \times C]$

Concepto
i_1 (1,A)
i_2 (2,A)
i_3 (3,B)
i_4 (4,B)

Fuente: propia del estudio.

Variables simbólicas

Una variable simbólica es una representación particular que permite agrupaciones por conceptos. Las variables se clasifican en multivaluada, intervalos e histograma.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Definición 8: Una variable multivaluada ***Y*** es aquella en la que los posibles valores que representa la variable simbólica son tomados del dominio del concepto. La lista completa de posibles valores es finita y los valores son definidos como valores no categóricos, es decir, cuantitativos. La tabla 5 muestra una columna llamada multivaluada, en donde por cada valor agrupado del concepto se cuenta la cantidad de veces que aparece dicho término en la agrupación.

Tabla 5
Ejemplo de $[A \times C]:\{multivaluada(B)\}$

Concepto	Multivaluada(B)
i_1 (1,A)	[34:2]
i_2 (2,A)	[34:1;36:1]
i_3 (3,B)	[34:1;31:1]
i_4 (4,B)	[36:1]

Fuente: propia del estudio.

Definición 9: Una variable intervalo es aquella cuyos valores son definidos como $\xi = [a,b] \subset \Re^1$, con $a \leq b, a, b \in \Re^1$, el intervalo puede ser abierto o cerrado $(a,b),[a,b],[a,b),(a,b]$. En la tabla 6 se observan los diversos intervalos, según la agrupación del concepto (Moore, 1979).

Tabla 6
Objeto simbólico de orden superior $[A \times C]:\{intervalo(B)\}$

Concepto	Intervalo(B)
i_1 (1,A)	[34.0,34.0]
i_2 (2,A)	[34.0,36.0]
i_3 (3,B)	[31.0,34.0]
i_4 (4,B)	[36.0,36.0]

Fuente: propia del estudio.

Definición 10: Una variable histograma ***Y*** toma posibles valores $\{n_k; k = 1, 2, \dots\}$ sobre el dominio $\mathcal{Y}$. Sea $\mathcal{Y}_{cat}$ el posible dominio de una variable multivaluada Y_{cat}, con $\mathcal{Y}_{cat} = \{n_1, n_2, \dots\}$, entonces, una variable modal multivaluada es aquella que toma sus valores del subconjunto de $\mathcal{Y}_{cat}$ con valores no negativos asociados a los valores del subconjunto, para una observación de una categoría toma la forma:

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

$$Y(w_u) = \xi_u = \{n_{u1}, \rho_{u1}; \ldots.; n_{us_u}, \rho_{us_u}\}$$

Donde $\{n_{u1}, \ldots., n_{us_u}\} \subseteq \mathcal{Y}_{cat}$ y donde n_{uk} aparece como el peso ρ_{uk}, $k = 1, \ldots., S_u$ y con $\sum_{k=1}^{s_u} p_{uk} = 1$. En la tabla 7 se muestran las distintas frecuencias de aparición del término.

Tabla 7
Objeto simbólico de orden superior $[A \times C]: \{intervalo(B)\}$

Concepto	histograma(B)
i_1 (1,A)	[34:1.0]
i_2 (2,A)	[34:0.5;36:0.5]
i_3 (3,B)	[34:0.5;31:0.5]
i_4 (4,B)	[36:1.0]

Fuente: propia del estudio.

Extensión simbólica

Debido a que los sistemas gestores de bases de datos no poseen las funcionalidades requeridas para crear, almacenar y manipular los objetos simbólicos, se propone implementar la extensión simbólica al lenguaje SQL para el sistema gestor de base de datos libre *PostgreSQL* (The PostgreSQL Global Development Group, 2012), el cual permite hacer modificaciones en sus programas fuentes. Una extensión, en general, para una base de datos, amplía el lenguaje SQL con nuevas directrices, tipos de datos, operadores y funciones. Para comprender mejor la dimensión de la extensión se muestra en la figura 1 su arquitectura conceptual.

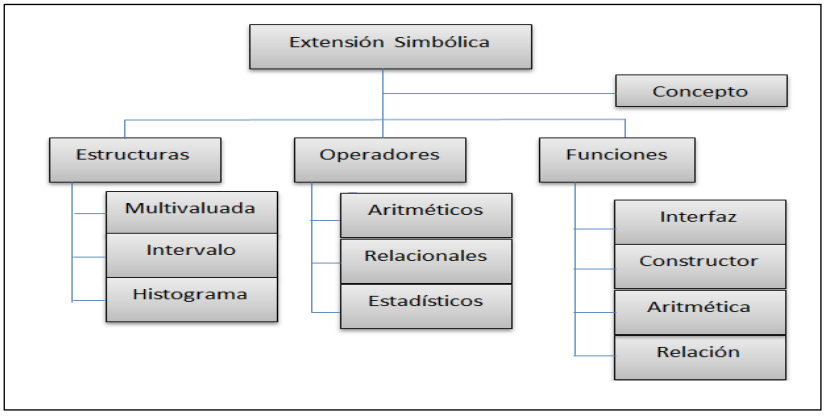

Figura 1.
Arquitectura conceptual de la extensión simbólica.
Fuente: propia del estudio.

La arquitectura conceptual de la extensión establece tres grupos de componentes. El primer, constituido por las estructuras necesarias de datos para registrar en la base de datos las variables simbólicas. Cada estructura se crea como un tipo de dato base, esto quiere decir que formará parte de los tipos primitivos del sistema gestor de base de datos. Los nuevos datos requieren de operadores aritméticos, operadores relacionales y operadores estadísticos. Las funciones en la extensión se clasifican en:

1. Interfaz: Funciones para establecer los formatos de entrada y salida del tipo de dato.
2. Constructor: Construyen los diferentes tipos de datos realizando agrupaciones.
3. Aritméticas: Permiten construir los operadores aritméticos para los diferentes tipos de datos.
4. Relación: Permiten construir los operadores relacionales para los diferentes tipos de datos.

En la tabla 8 se muestra en detalle los tipos de datos, los operadores y funciones para los tipos de datos intervalo.

Tabla 8
Componentes de la extensión simbólica para tipos intervalo

Prototipo de función	Descripción
TYPE sint as (x float, y float);	Construye un tipo intervalo conformado
AGGREGATE sint (float)	Constructor de intervalo
Funciones aritméticas para operadores	
FUNCTION addSInt(a sint, b sint)	Suma dos intervalos
FUNCTION lessSInt(a sint, b sint)	Resta dos intervalos
FUNCTION divInt(a sint, b sint)	Divide dos intervalos
FUNCTION mvlInt(a sint, b sint)	Multiplica dos intervalos
Funciones para operadores estadísticos	
FUNCTION abs(a sint)	Calcula el valor absoluto de un intervalo
FUNCTION w(a sint)	Calcula la media de un intervalo
FUNCTION m(a sint)	Calcula el promedio de la media de un intervalo
Funciones para operadores relacionales	
FUNCTION equal(a sint, b sint)	Determina si dos intervalos son equivalentes
FUNCTION different(a sint, b sint)	Determina si dos intervalos son distintos
FUNCTION greaterThan(a sint, b sint)	Determina si el primer intervalo es mayor que el segundo intervalo

Fuente: propia del estudio.

Se crean, además de las funciones para los tipos de datos, otras funciones para construir los objetos simbólicos que transforman las variables seleccionadas a conceptos y su intensión. La función symbolic_object() es la encargada de la transformación. El algoritmo general se muestra en la siguiente figura 2.

PASO	***Symbolic_object (Tabla, concepto, variable, [aserción])***
1	*Se crea una tabla de datos clásicos* *Sea* $T = Tabla$
2	*Se eliminan registros de T según aserción* $T = T - A, A = \{aserción(T)\}$
3	*Para cada una de las variables* C_i *del concepto* $C = C_1 \times C_2 \times \ldots \times C_n$
4	*Se construye una tabla intermedia formada por el concepto y los datos* $Stable = C \cup T$
5	*Sea* T_{salida} *= Tabla*
6	*Se crea una partición (agrupación) tal que* $Stable = P_1 \cup P_2 \cup \ldots \cup P_n$ $P = \{P_i : i \in Stable\}$
7	*Por cada* V_i *de las variables se aplica sobre* P_j *la transformación* $T_{salida} = \bigcup variable(P_j, V_i)$
8	*Se ordena y une con el concepto* $T_{salida} = C \cup T_{salida}$
9	*Fin del algoritmo* $Retornar(T_{salida})$

Figura 2.
Algoritmo para construcción de datos simbólicos.
Fuente*:* propia del estudio.

Pruebas funcionales a la extensión simbólica

Para realizar las pruebas de funcionalidad se ha creado una tabla relacional basada en el ejemplo *Medical* (Billard, 2006) y extendida a 600.000 observaciones. En esta tabla se han registrado las variables médicas de exámenes aplicados a personas de diferente sexo y edad, con datos de cirugías realizadas en ellos. Las pruebas tienen el objetivo de corroborar el funcionamiento y ejemplificar el uso sintáctico de la gramática de la extensión. En la tabla 9 se especifica el esquema de la tabla *Medical*.

Tabla 9
Tabla relacional Medical

Nombre de la columna	Tipo de dato	Detalle
Tipo	CHAR(1)	O,M,D
Sexo	CHAR(1)	Sexo (F=femenino, M=masculino)
Edad	DOUBLE	Edad entre 6 y 96
Estado	TEXT	Estado (S=soltero, C=casado)
Padres	DOUBLE	Número de padres
Peso	DOUBLE	Peso
Pulso	DOUBLE	Pulso
Psistólica	DOUBLE	Presión sistólica
Pdiastólica	DOUBLE	Presión diastólica
Colesterol	DOUBLE	Colesterol

Nota: Symbolic Data Analysis: Conceptual Statistics And Data Mining (Billard, 2006).
Fuente*:* propia del estudio.

Ejemplo 1: Construcción de un concepto. Utilizando la tabla *medical* se construye un concepto con las categorías de sexo y edad del paciente. Formalmente el concepto del ejemplo se representa por $s = [sexo \times edad]$. En el código 1 se crea el concepto s.

Código 1. Creación de un concepto

```
select create_symbolic_object ('medical','sexo,edad');
```

Como se aprecia en el código 1, es necesario invocar la función *create_symbolic_object*, la cual recibe como primer parámetro el nombre de la tabla relacional (origen de datos) y como segundo parámetro los conceptos separados por comas ",". En la tabla 10 se aprecia la ejecución de una vista llamada *stable* creada por la función *create_symbolic_object* del código 1.

Tabla 10
Construcción de un concepto $s = [sexo \times edad]$

concept text	tipos char	sexo charac	edad double p	estado text	padre doubl	peso double p	pulso double p	psistolic double p	pdiastol double p	coleste double
(F,6)	D	F	6	S	0	87	72	192	98	127
(F,6)	D	F	6	S	0	65	84	187	96	127
(F,6)	D	F	6	S	0	36	82	175	92	127
(F,6)	D	F	6	S	0	86	85	168	101	127
(F,6)	D	F	6	S	0	64	72	174	84	127
(F,6)	D	F	6	S	0	86	85	194	118	127
(F,6)	D	F	6	S	0	55	82	181	86	127
(F,6)	D	F	6	S	0	54	82	176	100	127
(F,6)	D	F	6	S	0	75	72	132	80	127
(F,6)	D	F	6	S	0	46	82	177	102	127
(F,7)	D	F	7	S	0	44	82	181	85	128
(F,7)	D	F	7	S	0	82	72	191	102	128

Fuente*:* propia del estudio.

Ejemplo 2: Construcción de una variable simbólica de tipo intervalo. En este ejemplo interesa analizar los mínimos y máximos de la columna peso, estos valores deben ser agrupados por el concepto de segundo grado "sexo,edad". Formalmente se conforma el objeto simbólico:

$$s = [sexo \times edad]\{int(peso), int(pulso)\}$$

El objeto simbólico anterior se construye haciendo uso del código 2, en la tabla 11 se muestra el resultado de su ejecución.

Código 2. Creación de un intervalo simbólico

```
SELECT create_symbolic_object( 'medical', 'gender,age',
'SINT(PESO),SINT(PULSO)');
```

La ejecución del código 2 genera la tabla 11, donde la segunda columna es una variable intervalo simbólico (pesos mínimos y máximos de la categoría sexo y edad), de igual forma la tercera columna representa un intervalo simbólico para el pulso.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Tabla 11
Resultado de la construcción de intervalos

concept text	sint sint	sint sint
(F,83)	(31,107)	(78,95)
(M,28)	(25,97)	(73,92)
(F,31)	(36,101)	(72,95)
(M,33)	(26,112)	(74,94)
(M,73)	(35,105)	(74,94)
(M,89)	(25,103)	(75,94)
(F,28)	(33,96)	(72,93)
(F,60)	(28,106)	(72,93)
(F,48)	(28,105)	(72,93)
(M,49)	(26,108)	(75,92)
(F,63)	(36,107)	(72,95)
(F,40)	(29,104)	(72,95)

Fuente*:* propia del estudio.

Ejemplo 3: Construcción de una aserción. Algunas veces, con el fin de disminuir la población de datos, es conveniente aplicar reglas lógicas sobre los registros $\Omega. A$ estas reglas se les llama una ***aserción***. Este ejemplo particular aplica una aserción sobre todos los registros de *Medical* tales que la edad del paciente sea mayor a 93 años. La implementación de esta aserción se logra mediante la ejecución del código 3. Formalmente se define el objeto simbólico:

$$s = [sexo, edad]\{sint(edad/edad > 93)\}$$

Código 3. Creación de una aserción

```
SELECT create_symbolic_object('medical', 'gender,age', 'sint(age)', 'age >= 93');
```

Como resultado de la ejecución del código anterior, se construye la tabla 12 con el tipo de datos intervalo y su aserción. El resultado de la ejecución es la generación de una tabla con cinco registros nuevos, de tipo intervalos degenerados[1].

[1] Un intervalo es degenerado si tiene la forma $\xi = [a, a]$ (Moore, 1979)

Tabla 12
Resultado de crear una aserción

	concept text	sint sint
1	(M,96)	(96,96)
2	(M,94)	(94,94)
3	(M,95)	(95,95)
4	(F,93)	(93,93)
5	(M,93)	(93,93)

Fuente*:* propia del estudio.

Ejemplo 4: Uso de operadores aritméticos para intervalos

Por otro lado, es necesario poder realizar operaciones aritméticas como sumar o restar intervalos. El objeto simbólico formado por la suma de dos intervalos simbólicos se define formalmente como:

$$s = [sexo \times edad]\{int(peso) + int(pulso)\}^2$$

Código 4**.** Operaciones aritméticas sobre intervalos

```
SELECT CONCEPT, SINT(PESO)+SINT(PULSO) FROM STABLE;
```

El resultado de ejecutar el código 4 genera la tabla 13, donde la segunda columna es el resultado de sumar el intervalo de peso más el intervalo de pulso. En la tabla 14 se tiene la lista completa de funciones aritméticas que soporta le extensión simbólica para los intervalos.

Tabla 13
Resultado de ejecutar una suma entre intervalos

concept text	?column? sint
(F,83)	(109,202)
(M,28)	(98,189)
(F,31)	(108,196)
(M,33)	(100,206)
(M,73)	(109,199)
(M,89)	(100,197)
(F,28)	(105,189)
(F,60)	(100,199)
(F,48)	(100,198)
(M,49)	(101,200)
(F,63)	(108,202)
(F,40)	(101,199)

Fuente*:* propia del estudio.

[2] Se define la suma de intervalos para $\xi_1 = [a,b]$ y $\xi_2 = [x,y]$, como $\xi_1 + \xi_2 = [a+x, b+y]$

Tabla 14
Operadores aritméticos para intervalos

Función	Símbolo en postgresql
Suma entre intervalos	+
Resta entre intervalos	-
Complemento de un intervalo	-
Multiplicación entre intervalos	*
División entre intervalos	/

Fuente: propia del estudio.

Ejemplo 5: Uso de operadores relacionales para intervalos. De la misma forma, es útil poder comparar intervalos en términos de su relación de orden[3]. Suponiendo que un investigador necesita determinar si los intervalos de la presión sistólica son mayores o iguales que los intervalos de la presión diastólica, dicha intensión de investigación puede ser resuelta aplicando el código 5, dejando el resultado en la tabla 15.

Código 5. Operaciones relacionales sobre intervalos

```
select concept, sint(psistolica), sint(pdiastolica),
sint(psistolica)>sint(pdiastolica) from stable;
```

Tabla 15
Resultado de ejecutar operaciones relacionales

concept text	sint sint	sint sint	?column? boolean
(F,83)	(118,194)	(70,114)	t
(M,28)	(124,197)	(76,121)	t
(F,31)	(123,194)	(71,110)	t
(M,33)	(123,197)	(73,117)	t
(M,73)	(130,197)	(74,120)	t
(M,89)	(122,196)	(73,117)	t
(F,28)	(132,194)	(70,110)	t
(F,60)	(125,193)	(72,111)	t
(F,48)	(121,194)	(70,111)	t
(M,49)	(125,197)	(73,116)	t
(F,63)	(126,192)	(71,116)	t
(F,40)	(123,191)	(75,115)	t

Fuente: propia del estudio.

[3] Se define para los intervalos $\xi_1 = [a, b], \xi_2 = [x, y]$ la relación de orden $\xi_1 < \xi_2 \Rightarrow b < x$ (Moore, 1979)

La cuarta columna de la tabla 15 representa una variable booleana, siendo t (true, verdadero) como resultado de la operación de comparar el primer intervalo contra el segundo. En la tabla 16 detalla la lista de los operadores relacionales de intervalos.

Tabla 16
Operadores relacionales para intervalos

Función	Símbolo en postgresql
Igual	=
Diferente	<>
Mayor	<
Mayor o igual	<=
Menor	>
subconjunto	Subset (….,….)

Fuente*:* propia del estudio.

Ejemplo 6: Uso de operadores conjuntistas para intervalos. De la misma forma como se desarrollaron las funciones aritméticas y relacionales, la extensión también contempla las operaciones conjuntistas. Este ejemplo aplica la unión[4] de intervalos, en la tabla 17 se muestra la ejecución del código 5. La tabla 18 se detalla todas las operaciones conjuntistas implementadas.

Código 6. Operaciones de conjuntos sobre intervalos

```
SELECT concept,unionSint( sint(psistolica),
sint(pdiastolica))from stable group by concept;
```

[4] Se define la unión de intervalos como $\xi_1 \underline{\cup} \xi_2 = [\min(a,x), \max(b,y)]$ para intervalos de la forma $\xi_1 = [a,b]\ y\ \xi_2 = [x,y]$

Tabla 17
Resultado de ejecutar un operador de unión sobre intervalos

concept text	sint sint	sint sint	unionsint sint
(F,83)	(118,194)	(70,114)	(70,194)
(M,28)	(124,197)	(76,121)	(76,197)
(F,31)	(123,194)	(71,110)	(71,194)
(M,33)	(123,197)	(73,117)	(73,197)
(M,73)	(130,197)	(74,120)	(74,197)
(M,89)	(122,196)	(73,117)	(73,196)
(F,28)	(132,194)	(70,110)	(70,194)
(F,60)	(125,193)	(72,111)	(72,193)
(F,48)	(121,194)	(70,111)	(70,194)
(M,49)	(125,197)	(73,116)	(73,197)
(F,63)	(126,192)	(71,116)	(71,192)
(F,40)	(123,191)	(75,115)	(75,191)

Fuente*:* propia del estudio.

Tabla 18
Operadores relacionales soportados para intervalos

Función	Símbolo en postgresql
Unión	unionSInt (…, ….)
Intersección	intersectSInt (…, ….)

Fuente*:* propia del estudio.

La extensión contempla otras funciones estadísticas, en la tabla 19 se observan el nombre de estas funciones implementadas para intervalos.

Tabla 19
Operadores estadísticos soportados para intervalos

Función	Símbolo en postgresql
Absoluto	Abs (…)
Distancia	W (…)
Media	M (…)

Fuente: propia del estudio.

Ejemplo 7: Construcción de una tabla simbólica. Una vez el investigador tenga la consulta SQL, la misma podrá ser guardada como una tabla simbólica mediante el código 7.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Código 7. Guardar una tabla simbólica

```
create table TablaEstudios as
select concept, sint(psistolica) col1, sint(pdiastolica)col2
from stable group by concept;
```

Herramienta de transformación de datos relacionales a datos simbólicos

En conjunto con la extensión simbólica *pSymbolic*, se desarrolló una aplicación informática para facilitar al usuario no experto la trasformación de datos relacionales a datos simbólicos. La misma puede ser descargada en la sección de *software* desde el sitio WEB www.slinfo.una.ac.cr. En la figura 3 se muestra la pantalla principal del aplicativo.

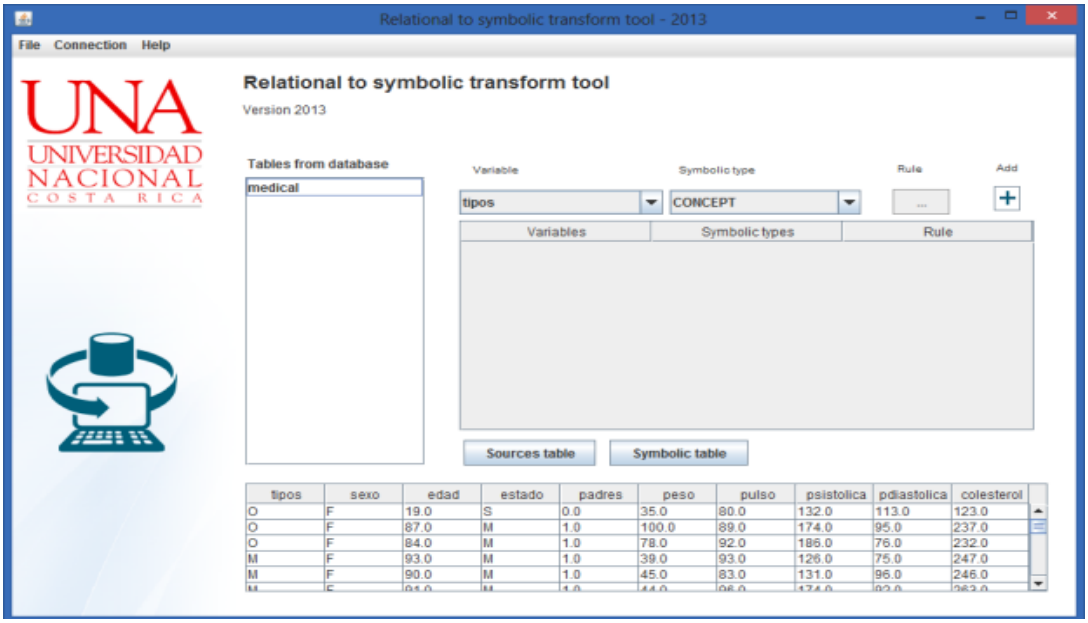

Figura 3.
Programa R2S, versión 1.0.
Fuente*:* propia del estudio.

El usuario después de escoger la fuente de datos puede observar los datos fuente mediante el botón "*Sources table*" y la tabla simbólica resultante con el botón "*Symbolic table*". Para crear un dato simbólico, la interfaz le ayuda establecer las aserciones y la definición del concepto (ver figura 4).

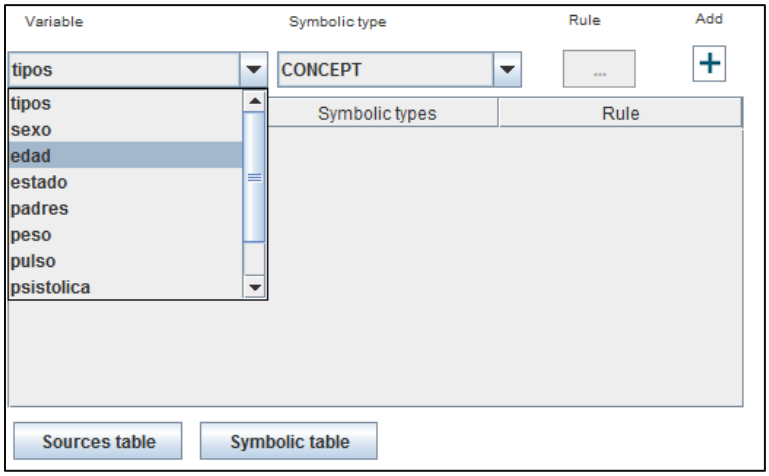

Figura 4.
Selección del concepto.
Fuente*:* propia del estudio.

Para crear las variables simbólicas, se debe escoger en primera instancia la columna de la tabla relacional y luego el tipo de variable simbólica en la que se va convertir, el proceso finaliza pulsando el botón "*Add*" (ver figura 5).

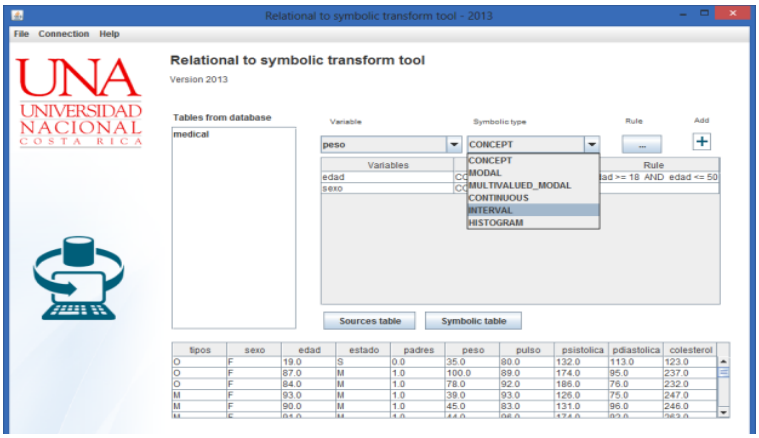

Imagen 5.
Variables simbólicas.
Fuente*:* propia del estudio.

Se termina el proceso al seleccionar la opción "*Symbolic table*", la cual despliega la información simbólica del siguiente objeto:

$$s = [sexo \times edad]\{int(peso)\}$$

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

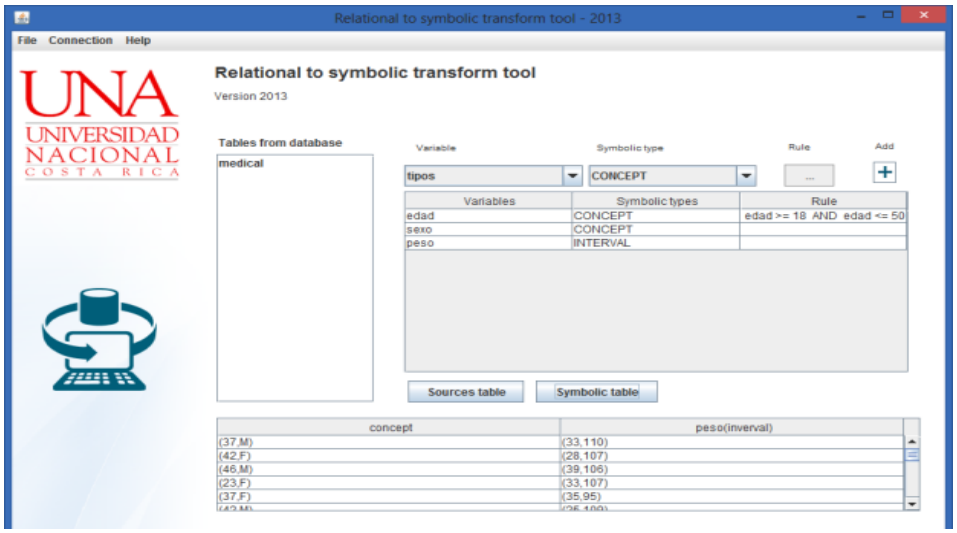

Figura 6.
Transformación simbólica, intervalo de peso.
Fuente: propia del estudio.

De forma similar se pueden agregar tantas variables simbólicas como el investigador requiera, para ejemplificarlo en la figura 7 se definieron dos datos simbólicos, la primera de tipo intervalo y la segunda un histograma. Formalmente se definió el objeto simbólico:

$$s = [sexo \times edad]\{int(peso), his(pulso)\}$$

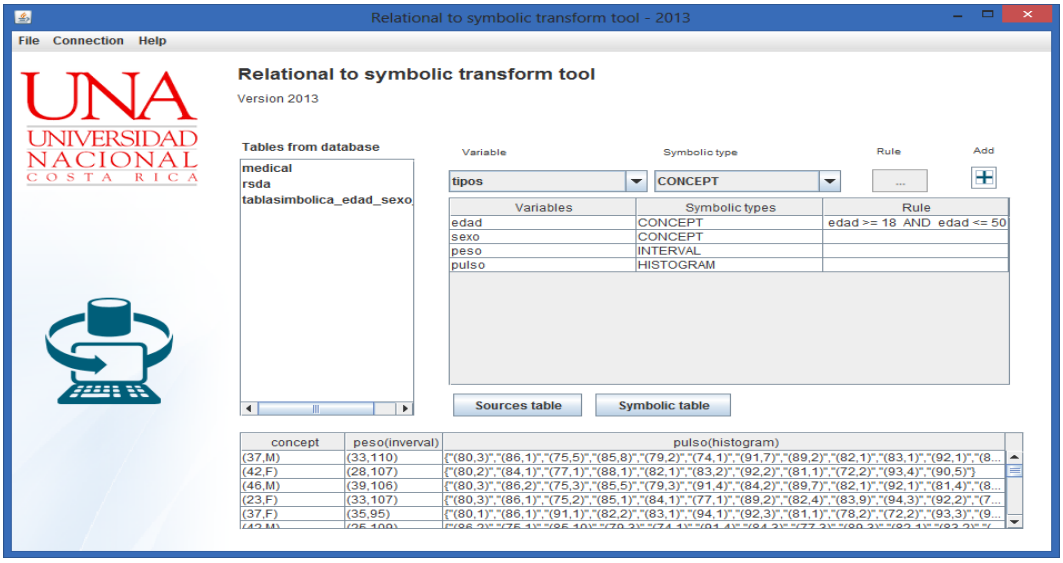

Figura 7.
Transformación simbólica, histograma de pulso.
Fuente: propia del estudio.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Utilización de la extensión

El software se hospedó en el portal WEB www.slinfo.una.ac.cr (ver figura 8) donde el lector encontrará información sobre la extensión. Esta página está dividida en tres secciones: introducción, documentación y descarga. La sección de introducción contiene una breve información sobre las funciones de la extensión sin entrar en detalles.

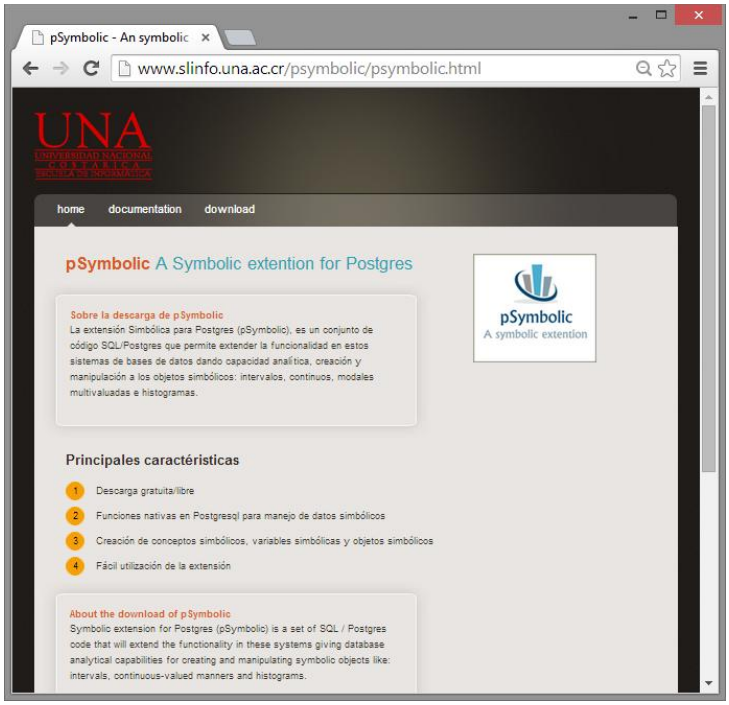

Figura 8.
Página principal de la extensión.
Fuente*:* propia del estudio.

La segunda opción muestra un manual de usuario en línea, donde se detallan los pasos para ejecutar las transformaciones simbólicas. Los ejemplos que se encuentran son:

1. Creación de un concepto.
2. Construcción de una variable simbólica de tipo intervalo.
3. Construcción de una aserción.
4. Uso de operadores aritméticos para intervalos.
5. Uso de operadores aritméticos para intervalos.
6. Uso de operadores conjuntistas para intervalos.
7. Construcción de una tabla simbólica.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

Figura 9.
Documentación de la extensión del sitio WEB.
Fuente*:* propia del estudio.

La pestaña de descargas permite obtener la extensión y la interfaz de trasformación de forma libre.

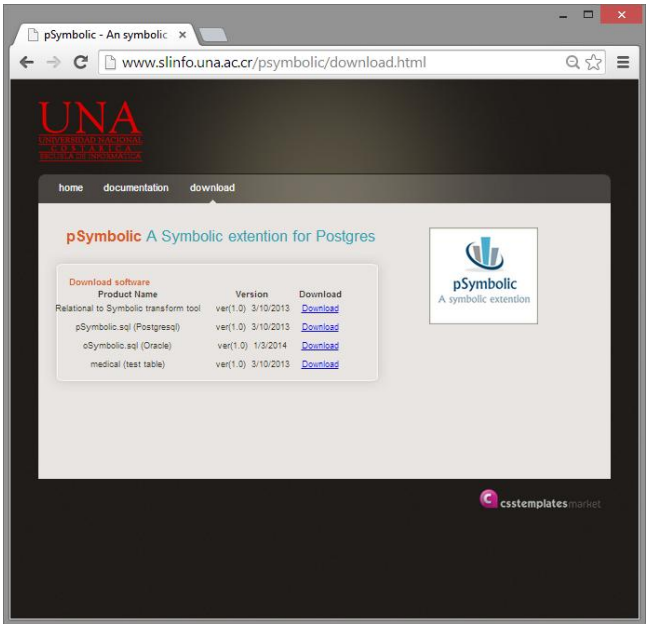

Figura 10.
Opción de descarga de la extensión.
Fuente*:* propia del estudio.

Uniciencia. Vol. 29, No. 2, pp. 15-38. Julio-Diciembre, 2015. ISSN Electrónico: 2215-3470
URL: www.revistas.una.ac.cr/uniciencia DOI: http://dx.doi.org/10.15359/ru.29-2.2
Email: revistauniciencia@una.cr

La extensión se implementó para los sistemas gestores de bases de datos *PostgreSql* y *Oracle*, permitiendo así que diferentes usuarios puedan utilizar la extensión y dando la posibilidad que los objetos simbólicos sean de libre utilización y permitanla posibilidad de incorporarlos en diferentes áreas de investigación y no solamente en el análisis de datos simbólicos.

Desde su publicación en el portal del Laboratorio de Bases de Datos de la Universidad Nacional a la fecha de esta publicación, se han descargado un total de 109 extensiones en los dos últimos meses. Los países que han realizo descarga del software se detallan en la imagen 11.

	Sesiones
[CURSO] Administración de Bases de Datos (Laboratorio de Bases de datos UNA UA-21078955-13)	57
[CURSO] Diseño e Implementación de BD (Laboratorio de Bases de datos UNA UA-21078955-12)	233
Metodología para la Administración de Bases de Datos Relacionales (Laboratorio de Bases de datos UNA UA-21078955-6)	2.314
[HERRAMIENTA] Manuel de Calidad Postgresql (Laboratorio de Bases de datos UNA UA-21078955-14)	102
Todos los datos de sitios web (Laboratorio de Bases de datos UNA UA-21078955-15)	9
pSymbolic (Laboratorio de Bases de datos UNA UA-21078955-10)	109
Todos los datos de sitios web (Laboratorio de Bases de datos UNA UA-21078955-10)	109
Sitio de descargas del RAT (Laboratorio de Bases de datos UNA UA-21078955-8)	7.399
Relational Algebra Translator (Laboratorio de Bases de datos UNA UA-21078955-7)	14.223
Sitio SLINFO completo (Laboratorio de Bases de datos UNA UA-21078955-11)	1.037

a) Visitas al Laboratorio de Bases de Datos

País/territorio	Adquisición Sesiones
	109 % del total: 100,00% (109)
1. Costa Rica	98 (89,91%)
2. Mexico	5 (4,59%)
3. Tunisia	3 (2,75%)
4. Argentina	1 (0,92%)
5. Peru	1 (0,92%)
6. United States	1 (0,92%)

b) visitas por país al sitio de la extensión

Figura 11.
Descargas de la extensión por país. Google Analytics a la fecha 26 de mayo del 2014.
Fuente*:* propia del estudio.

Conclusiones

Con la finalización de esta investigación, se desarrollaron una serie de productos publicados en los sitios oficiales de la Universidad Nacional, que permitieron aumentar la visibilidad de las investigaciones realizadas por la Escuela de Informática a nivel global.

De esta manera se publica, de forma gratuita, una extensión para el manejo de los datos simbólicos que son utilizables tanto para el sistema gestor Oracle como PostgreSQL. Esta extensión contempla funciones de construcción, visualización, aritméticas, estadísticas y lógicas.

Por otro lado, se facilita la descarga de una herramienta para el manejo de la extensión para aquellos usuarios de perfil no técnico. En consecuencia, se espera que el dato simbólico se conozca aún más al extender la tecnología de bases de datos en función de necesidades reales de los investigadores.

El proyecto de investigación 0281-13, "Implementación de extensiones simbólicas al lenguaje SQL en servidores de bases de datos objeto-relacionales" proyecto suscrito al Área de Investigación y formulado por el investigador Johnny Villalobos Murillo de la Escuela de Informática y Computación de la Facultad de Ciencias Exactas y Naturales de la Universidad Nacional de Costa Rica y sus colaboradores, contribuye a disminuir la brecha tecnológica, el acceso libre a las tecnología de información y el quehacer de esta Universidad en función de la sociedad.

La tecnología desarrollada puede ser un marco metodológico de apoyo a investigaciones en bases de datos. La creación de extensiones puede formar parte de contenidos de cursos avanzados en bases de datos. La extensión simbólica es una forma de reducir grandes volúmenes de datos tal y como se demostró en las pruebas funcionales, haciendo más manipulables los datos para el análisis de datos simbólicos mediante el uso de los nuevos operadores simbólicos.

Referencias.

Billard, L. (2006). *Symbolic Data Analysis: Conceptual Statistics And Data Mining.* New York: Wiley. DOI http://dx.doi.org/10.1002/9780470090183

Bock, H-H., & Diday, E. (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. *Springer Verlag*, 425.

Codd, E. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 377-387. DOI http://dx.doi.org/10.1145/362384.362685

Diday, E. (2009). The state of the art in symbolic data analysis: overview and future. Obtenido de http://media.wiley.com/product_data/excerpt/36/04700188/0470018836-1.pdf

Moore, R. (1979). Methods and Applications of Interval Analysis. Philadelphia, USA: Society for Industrial and Applied Mathematics (SIAM)

Rodríguez, O. (2000). The Knowledge Mining Suite (KMS). San José, Costa Rica: University of Costa Rica and Predisoft International S.A

The PostgreSQL Global Development Group. (2012). PostgreSQL Developer's Guide. Obtenido de http://www.postgresql.org