Especificación de Primitivas en SQL

Grupo de Concepción de Sistemas de Información,
InCo, Facultad de Ingeniería, Universidad de la República
Montevideo, Uruguay (http://www.fing.edu.uy/~csi/)
Ignacio Larrañaga (http://www.isoft.com.uy/~il/)

Abstract

En este articulo se trata el problema de expresar las primitivas expuestas en la tesis de maestría *"Data Warehouse Design and Maintenance through Schema Transformations"* en SQL estándar. Esto quiere decir que las transformaciones que dicha primitiva aplica a las instancias son expresables utilizando solamente sentencias SQL exclusivamente.

1- Introducción

El objetivo de esta investigación se centra en expresar las operaciones necesarias para la transformación de las instancias al aplicar las primitivas presentadas en el trabajo de tesis [Am2000].
El interés principal de esta investigación es que el resultado va a ser utilizado para plantear estrategias de carga y actualización de la información del DW ("Data Warehouse") que se plantea en dicha tesis.

En el resto del articulo procederemos de la siguiente forma; en la sección 2 a dar contexto recordando las primitivas a utilizar y otros aspectos útiles. En la sección 3 haremos el análisis de las primitivas, dividiendo en AR ("Álgebra Relacional") primeramente y posteriormente SQL. Continuaremos mencionando los trabajos futuros en la sección 4. En la sección 5 daremos las conclusiones del análisis y concluiremos mencionando la bibliografía utilizada.

2- Contexto

Nos referiremos aquí a las primitivas de diseño que se tratan en la tesis citada anteriormente, por lo cual las recordaremos a continuación(1).

- **P1 Identity.** Dada una relación esta genera otra que es exactamente la misma que la relación origen.
- **P2 Data Filter.** Dada una relación origen, esta genera otra donde solo se preservan algunos atributos. El objetivo es eliminar los atributos puramente operacionales.
- **P3 Temporalization.** Esta agrega un elemento de tiempo al conjunto de atributos de la relación.
- **P4 Key Generalization. (*)** Estas primitivas generalizan la clave de una relación de dimensión, entonces mas de una tupla por cada elemento de la relación puede ser almacenado.
- **P5 Foreign Key Update.** A través de esta primitiva una clave foránea y sus referencias pude ser cambiada en una relación. Esto es muy usado cuando las claves primarias son modificadas.
- **P6 DD-Adding. (*)** Las primitivas de este grupo agregan a una relación un atributo que es derivado de otros.
- **P7 Attribute Adding.** Esta primitiva agrega atributos a la relación de dimensión. Esto puede ser muy útil para mantener en la misma tupla mas de una versión de un atributo.
- **P8 Hierarchy Roll Up.** Esta primitiva realiza el "roll up" por un atributo de una relación siguiente la jerarquía. Además esta puede generar otra relación de jerarquía con el nivel de detalle correspondiente.
- **P9 Aggregate Generation.** Dada una relación de medida, esta primitiva genera otra relación de medida, donde la información es resumida (o agrupada) por un conjunto dado de atributos.
- **P10 Data Array Creation.** Dada una relación que contiene un atributo de medida y un atributo que representa un conjunto predeterminado de valores, esta primitiva genera una relación con la información estructurada como array.

[1] La información siguiente simplemente fue traducida para ser extraída del documento. Los asteriscos en la definición denotan que se esta hablando acerca de un grupo de primitivas.

- **P11 Partition by Stability. (*)** Estas primitivas particionan una relación, con el objetivo de organizar el almacenamiento histórico de la información. Particiones Horizontales o Verticales pueden ser aplicadas dependiendo del criterio de diseño utilizado.
- **P12 Hierarchy Generation. (*)** Esta es una familia de primitivas que genera relaciones de jerarquía tomando como entrada relaciones que incluyen una jerarquía o parte de una.
- **P13 Minidimension Break off.** Esta primitiva elimina un conjunto de atributos de una relación de dimensión, construyendo una nueva relación con ellos.
- **P14 New Dimension Crossing.** Esta primitiva permite materializar un cruzamiento de dimensiones en una nueva relación.

Tomaremos las mismas definiciones básicas que en [Am2000] sobre conjuntos de relaciones y conjuntos de relaciones, las citaremos en el "Apéndice 1".

Introduciremos los siguientes conceptos:

- **Expresión de Agregación**: Esta es una expresión formada por alguno de los operadores de agregación definidos en [ElNa1997] (SUM, COUNT, etc.).

Introduciremos los siguientes dominios:

- **Time**: Dominio de los valores de tiempo, contiene valores que nos permiten registrar un instante de tiempo, p.e.: "23/4/2001" $\in$ Time.

3- Análisis

Tomaremos cada una de las primitivas definidas y haremos una propuesta para la sentencia SQL. Y para las que sea necesario modificar la definición de alguna relación, daremos los cambios necesarios.

- **P1 Identity.**

Entrada:

Esquema origen: $R \in Rel$

Procesamiento:

P1 = select * from R

- **P2 Data Filter.**

Entrada:

Esquema origen: $R(A_1, ..., A_n) \in Rel$

Conjunto de atributos a filtrar: $X \subset \{A_1, ..., A_n\}$

Pre-procesamiento:

$A' = \{A'_1, ..., A'_m\} = \{A_1, ..., A_n\} - X$

Procesamiento:

P2 = select $A'_1, ..., A'_m$ from R

- **P3 Temporalization.**

Entrada:

Esquema origen: $R \in Rel$

Valor de tiempo: $t \in Time$

Procesamiento:

P2 = select *, t from R

- **P4 Key Generalization.**
- **P4.1 Version Digits.**

Entrada:

Esquema origen: $R(A_1, ..., A_n) \in Rel$ / $A_1 \in X \in Att_k(R)$

Numero de versión: $n \in String$.

Procesamiento(2):

P4.1 = select n || $A_1, ..., A_n$ from R

- **P4.2 Key Extension.**

Entrada:

Esquema origen: $R \in Rel$

Procesamiento:

P1 = select * from R

- **P5 Foreign Key Update.**

Entrada:

Esquema origen: $R(A_1, ..., A_n) \in Rel$

Esquema de la clave foránea: $T(B_1, ..., B_m) \in Rel$

Antigua clave foránea: $X \subseteq \{A_1, ..., A_n\}$

Nueva clave foránea: $Y \subseteq \{B_1, ..., B_m\}$

Esquema de correspondencias: $S(C_1, ..., C_i) \notin Rel$, $\{C_1, ..., C_i\} = (X \cup Y)$

Pre-procesamiento:

$V = \{V_1, \ldots, V_j\}$ / $V = Y \cup (\{A_1, ..., A_n\} - X)$

Procesamiento:

P5 = select $V_1, \ldots, V_j$ from R, S where R.X = S.X

- **P6 DD-Adding.**
- **P6.1 DD-Adding 1-1.**

Entrada:

Esquema origen: $R(A_1, ..., A_n) \in Rel$

Función de calculo: $f(X)$ / $X \subseteq \{A_1, ..., A_n\}$

Procesamiento:

P6.1 = select *, f(X) from R

- **P6.2 DD-Adding N-1.**

Entrada:

[2] El símbolo "||" hace referencia al operador de concatenación, tal y como se define en [ElNa1997].

Esquemas origen: $R(A_1, ..., A_n)$, $R_1(A'_1, ..., A'_{n'})$, ..., $R_n(A^n{}_1, ..., A^n{}_n n) \in Rel$
Función de calculo: $f(X) / X \subseteq (\{A_1, ..., A_n\} \cup \{A'_1, ..., A'_{n'}\} \cup ... \cup \{A^n{}_{1, ...,} A^n{}_n n\})$
Atributos de Join: $Y \subseteq \{A_1, ..., A_n\} \wedge Y' \subseteq \{A'_1, ..., A'_{n'}\} \wedge ... \wedge Y^n \subseteq \{A^n{}_1, ..., A^n{}_n n\}$

Procesamiento:
P6.2 = select $A_1, ..., A_n, f(X)$
from $R, R_1, R_2, ..., R_n$
where $R.Y = R_1.Y'$ and $R_1.Y' = R_2.Y^2$
and … and $R_{(n-1)}.Y^{(n-1)} = R_n.Y^n$

- **P6.3 DD-Adding N-N.**

Entrada:
Esquemas origen: $R(A_1, ..., A_n)$, $R_1(A'_1, ..., A'_{n'})$, ..., $R_n(A^n{}_1, ..., A^n{}_n n) \in Rel$
Expresión de agregación: $e(X) / X \in (\{A'_1, ..., A'_{n'}\} \cup ... \cup \{A^n{}_{1, ...,} A^n{}_n n\})$
Atributos de Join: $Y \subseteq \{A_1, ..., A_n\} \wedge Y' \subseteq \{A'_1, ..., A'_{n'}\} \wedge ... \wedge Y^n \subseteq \{A^n{}_1, ..., A^n{}_n n\}$
Atributos de agregación: $Z \subseteq (\{A'_1, ..., A'_{n'}\} \cup ... \cup \{A^m{}_{1, ...,} A^m{}_n n\})$

Procesamiento:
P6.3 = select $A_1, ..., A_n, e(X)$
from $R, R_1, R_2, ..., R_n$
where $R.Y = R_1.Y'$ and $R_1.Y' = R_2.Y^2$
and … and $R_{(n-1)}.Y^{(n-1)} = R_n.Y^n$
group by $A_1, ..., A_n, Z$

- **P7 Attribute Adding.**

Entrada:
Esquema origen: $R \in Rel$
Valores de los atributos a agregar: $\{b_1, ..., b_n\}$

Procesamiento:
P7 = select *, $b_1, ..., b_n$ from R

- **P8 Hierarchy Roll Up.**

Entrada:
Esquemas origen:
$R_1(A_1, ..., A_n) \in Rel_M / \exists A \subset \{A_1, ..., A_n\} \wedge A \subset Att_{FK}(R_1, R_2)$
$R_2(B_1, ..., B_{n'}) \in Rel_J / A \subset \{B_1, ..., B_{n'}\} \wedge A \subset Att_K(R_2)$
Atributos de medida: $Z = \{Z_1, ..., Z_k\} = Att_M(R_1), Z \subset \{A_1, ..., A_n\}$
Agregaciones de los atributos: $\{e_1(Z_1), ..., e_k(Z_k)\}$
Nivel de la jerarquía: $B / B \subset \{B_1, ..., B_{n'}\} \wedge B \subset Att_D(R_2)$
Atributos que por su granularidad salen de R_1: $X / X \subset \{A_1, ..., A_n\} \wedge X \subset (Att_D(R_1) \cup Att_M(R_1))$
Atributos que por su granularidad salen de R_2: $Y / Y \subset \{B_1, ..., B_{n'}\}$
Indica si genera nueva jerarquía: agg_h $\in$ Bolean

Pre-procesamiento:
$V = \{V_1, ..., V_m\} / V = ((((\{A_1, ..., A_n\} - A) \cup B) - X) - Z$
$V' = \{V'_1, ..., V'_{m'}\} / V' = \{B_1, ..., B_{n'}\} - Y$

Procesamiento:
P8 = select $V_1, ..., V_m, e_1(Z_1), ..., e_k(Z_k)$
from R_1, R_2
where $R_1.A = R_2.A$
group by $V_1, ..., V_m$

Si se indica agg_h:
P8b = select distinct $V'_1, ..., V'_{m'}$
from R_2

- **P9 Aggregate Generation.**

Entrada:
Esquemas origen: $R(A_1, ..., A_n) \in Rel_M$
Atributos de medida: $Z = \{Z_1, ..., Z_k\} = Att_M(R), Z \subset \{A_1, ..., A_n\}$
Agregaciones de los atributos: $\{e_1(Z_1), ..., e_k(Z_k)\}$
Atributos que salen de R: $X / X \subset \{A_1, ..., A_n\} \wedge X \subset (Att_D(R) \cup Att_M(R))$

Pre-procesamiento:
$V = \{V_1, ..., V_m\} / V = (\{A_1, ..., A_n\} - X) - Z$

Procesamiento:
P9 = select $V_1, ..., V_m, e_1(Z_1), ..., e_k(Z_k)$
from R
group by $V_1, ..., V_m$

- **P10 Data Array Creation.**

Entrada:
Esquema origen: $R(A_1, ..., A_n)$
Atributo de valores predefinidos: $A \in \{A_1, ..., A_n\}$
Expresión agregación: $e(A)$

Pre-procesamiento:
$V = \{V_1, ..., V_m\}$ /
V = select distinct A
from R
$B = \{B_1, ..., B_p\} = Att_M(R)$
$N = \{ N_{ij} / N_{ij} = \text{“}V_i\text{”} \| \text{“\_”} \| \text{“}B_j\text{”}, i=1..m, j=1..p\}$
$K = \{K_1, ..., K_{n-1}\} = \{A_1, ..., A_n\} - B - \{A\}$

Procesamiento:
T_1 = select $K_1, ..., K_{n-1}, e(B_1)$ as $N_{11}, ..., e(B_p)$ as N_{p1} from R where $A = V_1$ group by $K_1, ..., K_{n-1}$
…

T_m = select $K_1, \ldots, K_{n-1}$, $e(B_1)$ as N_{1m}, …, $e(B_p)$ as N_{pm} from R where A = V_m group by $K_1, \ldots, K_{n-1}$

P10 = select $K_1, \ldots, K_{n-1}, N_{11}, \ldots, N_{pm}$
from $T_1, \ldots, T_m$
where $T_1.K=T_2.K$ and…and $T_{m-1}.K=T_m.K$

- **P11 Partition by Stability.**
- **P11.1 Vertical Partition.**

Entrada:

Esquema origen: $R \in Rel$
Atributos que nunca cambian: $Y \subset Att(R)$
Atributos que algunas veces cambian: $Z \subset Att(R)$, $Z \cap Y = \varnothing$
Atributos que cambian muchas veces: $W \subset Att(R)$, $W \cap Y = \varnothing \wedge W \cap Z = \varnothing$

Pre-procesamiento:

$Y' = \{Y'_1, \ldots, Y'_{n'}\}$ / $Y' = Att_K(R) \cup Y$
$Z' = \{Z'_1, \ldots, Z'_{n'}\}$ / $Z' = Att_K(R) \cup Z$
$W' = \{W'_1, \ldots, W'_{n'}\}$ / $W' = Att_K(R) \cup W$

Procesamiento:

P11.1.1 = select $Y'_1, \ldots, Y'_{n'}$ from R
P11.1.2 = select $Z'_1, \ldots, Z'_{n''}$ from R
P11.1.3 = select $W'_1, \ldots, W'_{n'''}$ from R

- **P11.2 Horizontal Partition.**

Entrada:

Esquema origen: $R \in Rel$
Condición de Historizacion: $c(X)$ / $X \subset Att(R)$

Procesamiento:

P11.2.1 = select * from R where c(X)
P11.2.2 = select * from R where not(c(X))

- **P12 Hierarchy Generation.**
- **P12.1 De-Normalized Hierarchy Generation.**

Entrada:

Esquemas origen: $R_1, \ldots, R_n \in Rel$
Atributos de la Jerarquía: $J = \{J_1, \ldots, J_m\}$
Clave de la Jerarquía: $k \subseteq \{J_1, \ldots, J_m\}$

Pre-procesamiento:

$S' = \{S'_1, \ldots, S'_{n'}\} = (Att(R_1) - J) \cup k$
…
$S^n = \{S'_1, \ldots, S'_{nn}\} = (Att(R_n) - J) \cup k$
$s_{i(i+1)} = Att(R_i) \cap J \cap Att(R_{i+1})$, i=1..(n-1)

Procesamiento:

P13.0 = select distinct $J_1, \ldots, J_m$
from $R_1, \ldots, R_n$
where $R_1.s_{12} = R_2.s_{12}$ and … and $R_{(n-1)}.s_{(n-1)n} = R_n.s_{(n-1)n}$

P13.1 = select $S'_1, \ldots, S'_{n'}$
from R_1, P13.0
where $R_1.k$ = P13.0.k
…
P13.n = select $S'_1, \ldots, S'_{nn}$
from R_n, P13.0
where $R_1.k$ = P13.0.k

- **P12.2 Snowflake Hierarchy Generation.**

Entrada:

Esquemas origen: $R_1, \ldots, R_n \in Rel$
Conjunto ordenado de atributos de la Jerarquía: $J = \{J_1, \ldots, J_m\}$
Clave de la Jerarquía: $k \subseteq \{J_1, \ldots, J_m\}$

Pre-procesamiento:

$S' = \{S'_1, \ldots, S'_{n'}\} = (Att(R_1) - J) \cup k$
…
$S^n = \{S'_1, \ldots, S'_{nn}\} = (Att(R_n) - J) \cup k$
$s_{i(i+1)} = Att(R_i) \cap J \cap Att(R_{i+1})$, i=1..(n-1)

Procesamiento:

T1 = select distinct $J_1, \ldots, J_m$
from $R_1, \ldots, R_n$
where $R_1.s_{12} = R_2.s_{12}$ and … and $R_{(n-1)}.s_{(n-1)n} = R_n.s_{(n-1)n}$

P13.J1 = select distinct J_1, J_2
from $R_1, \ldots, R_n$
where $R_1.s_{12} = R_2.s_{12}$ and … and $R_{(n-1)}.s_{(n-1)n} = R_n.s_{(n-1)n}$
…
P13.J(m-1) = select distinct $J_{(m-1)}, J_m$
from $R_1, \ldots, R_n$
where $R_1.s_{12} = R_2.s_{12}$ and … and $R_{(n-1)}.s_{(n-1)n} = R_n.s_{(n-1)n}$

P13.1 = select $S'_1, \ldots, S'_{n'}$
from R_1, T1
where $R_1.k$ = T1.k
…
P13.n = select $S'_1, \ldots, S'_{nn}$
from R_n, T1
where $R_1.k$ = T1.k

- **P12.3 Free Decomposition – Hierarchy Generation**

Entrada:

Esquemas origen: $R_1, \ldots, R_n \in Rel$
Conjunto ordenado de atributos de la Jerarquía: $J = \{J_1, \ldots, J_m\}$
Descomposición de la Jerarquía: $D = \{D_i \,/\, D_i = \{J^i_1, \ldots, J^i_{qi}\} \subset J, i=1..p\}$
Clave de la Jerarquía: $k \subseteq \{J_1, \ldots, J_m\}$

Pre-procesamiento:

$S' = \{S'_1, \ldots, S'_{n'}\} = (Att(R_1) - J) \cup k$
…
$S^n = \{S'_1, \ldots, S'_{nn}\} = (Att(R_n) - J) \cup k$
$s_{i(i+1)} = Att(R_i) \cap J \cap Att(R_{i+1})$, i=1..(n-1)

Procesamiento:

T1 = select distinct $J_1, \ldots, J_m$
from $R_1, \ldots, R_n$
where $R_1.s_{12} = R_2.s_{12}$ and … and $R_{(n-1)}.s_{(n-1)n} = R_n.s_{(n-1)n}$
P13.J1 = select distinct $J'_1, \ldots, J'_{q'}$
from $R_1, \ldots, R_n$
where $R_1.s_{12} = R_2.s_{12}$ and … and $R_{(n-1)}.s_{(n-1)n} = R_n.s_{(n-1)n}$
…
P13.Jp = select distinct $J^p_1, \ldots, J^p_{qp}$
from $R_1, \ldots, R_n$
where $R_1.s_{12} = R_2.s_{12}$ and … and $R_{(n-1)}.s_{(n-1)n} = R_n.s_{(n-1)n}$
P13.1 = select $S'_1, \ldots, S'_{n'}$
from R_1, T1
where R_1.k = T1.k
…
P13.n = select $S'_1, \ldots, S'_{nn}$
from R_n, T1
where R_1.k = T1.k

- **P13 Minidimension Break off.**

Entrada:

Esquema origen: $R \in Rel$
Función de clave: f
Atributos de la mini dimensión: $X = \{X_1, \ldots, X_n\} \subset Att(R)$

Pre-procesamiento:

$\{R'_1, \ldots, R'_m\} = Att(R) - X$

Procesamiento:

T1 = select f as F, * from R
P13.1 = select F, $X_1, \ldots, X_n$ from T1
P13.2 = select F, $R'_1, \ldots, R'_m$ from T1

- **P14 New Dimension Crossing.**

Entrada:

Esquema origen: $R_1, R_2 \in Rel$
Atributos de Join: A, $A \subset Att(R_1)$, $A \subset Att(R_2)$
Atributos que se excluyen de R_1: $Y_1 \subset Att(R_1)$
Atributos que se excluyen de R_2: $Y_2 \subset Att(R_2)$

Pre-procesamiento:

$Y'_1 = \{y'_1, \ldots, y'_n\} = Att(R_1) - Y_1$
$Y'_2 = \{y''_1, \ldots, y''_m\} = Att(R_2) - Y_2$

Procesamiento:

P14 = select distinct $y'_1, \ldots, y'_n, y''_1, \ldots, y''_m$ from R_1, R_2 where $R_1.A = R_2.A$

4-Trabajos Relacionados

5- Trabajos Futuros

Dada la línea de trabajo de tratar de definir completamente las primitivas citadas en el documento, aparece como una posible línea de trabajo atacar el problema de definir formalmente su semántica.
Otro aspecto que resulta interesante estudiar tiene que ver con la completitud de estas primitivas. Obviamente ya es un problema definir completitud, tanto así lo será entonces demostrar que son completas.
También aparece a la luz la existencia de ciertas propiedades entre las primitivas que deberían ser estudiadas, p.e. al existir la identidad existe la propiedad de reflexión: P? . P1 = P? = P1 . P?.

6- Conclusión

Como conclusión podemos decir ..

7- Apéndices

<u>*Apéndice 1*</u>: Conjuntos de Relaciones y Atributos de [Am2000]

Conjuntos de Relaciones(1, 3):

- **Rel** Conjunto de todas las relaciones (cualquier tipo de relaciones).
- **Rel_D** Conjunto de relaciones de "dimensión". Estas son relaciones que representan información descriptiva acerca de los sujetos del mundo real.
- **Rel_C** Conjunto de relaciones de "cruzamiento". Estas son relaciones que representan relaciones o combinaciones entre elementos de un grupo de dimensiones. Usualmente, estas contienen atributos que representan medidas para las combinaciones.
- **Rel_M** Conjunto de relaciones de "medida". Estas son las relaciones de cruzamiento que tienen al menos un atributo de medida.
- **Rel_J** Conjunto de relaciones de "jerarquía". Estas son las relaciones de dimensión que contienen un conjunto de atributos que constituyen una jerarquía. El hecho de que exista una jerarquía entre un conjunto de atributos, solo puede ser determinado tomando en cuenta la semántica de estos.

[3] En este trabajo igual que en [Am2000] utilizaremos la palabra relación como sinónimo de esquema de relación.

- **Rel_H** Conjunto de relaciones "históricas". Estas son relaciones que tienen información histórica que se corresponde con información en otra relación. Nosotros definimos la función $f_H : Rel_H \rightarrow Rel$, la cual dada una relación histórica, retorna la relación correspondiente actual.

Conjuntos de Atributos[(4)]:
- **Att(R)** Conjunto de todos los atributos de la relación R.
- **Att_M(R)** Conjunto de atributos de medida de la relación R.
- **Att_D(R)** Conjunto de atributos descriptivos de la relación R.
- **Att_C(R)** Conjunto de atributos derivados (calculados) de la relación R.
- **Att_J** Conjunto de atributos que representan una jerarquía.
- **Att_K(R)** Conjunto de conjuntos de atributos que son clave en la relación R.
- **Att_{FK}(R)** Conjunto de conjuntos atributos que son clave foránea en una relación R.
- **Att_{FK} (R1, R2)** Conjunto de atributos que son clave foránea en una relación R1 con respecto a una relación R2.

8- Bibliografía

- **[Am2000]** Adriana Marotta, Data Warehouse Design and Maintenance through Schema Transformations., Grupo de Concepción de sistemas de información, (*http://www.fing.edu.uy/~csi/publicaciones/lista_pub_csi2000.html*, marzo 2001)
- **[ElNa1997]** R. Elmasri, S. B. Navathe, Sistemas de Bases de Datos, Conceptos Fundamentales (Segunda Edición). ISBN: 0-201-65370-2.

4 En algunos conjuntos se registraron cambios dado que se extendió su expresividad.