Enunciado

Se han propuesto dos claves candidatas A y CD para una relación R de esquema {A, B, C, D} que verifica el conjunto F de dependencias funcionales {A→B, C→B, BD→A, A→D, B→C}. ¿Son A y CD claves? ¿Son las únicas?

Resolución

Una forma de comprobar si las dos claves proporcionadas son las únicas e, incluso si son claves, consiste en aplicar el algoritmo de cálculo de claves candidatas.

En primer lugar, sería recomendable establecer los tipos de atributos en función de si aparecen a la izquierda en todas las dependencias funcionales en las que aparece, a la derecha en todas las dependencias funcionales en las que aparece, a la izquierda de alguna y a la derecha en otra o en ninguna dependencia funcional:

- atributos independientes (no aparecen en ninguna dependencia): ninguno
- atributos equivalentes: C→B y B→C establecen una equivalencia entre B y C
- atributos que aparecen a la izquierda en todas las dependencias en las que aparecen: ninguno
- atributos que aparecen a la izquierda de alguna dependencia funcional y a la derecha de otra: {A, B, C y D}
- atributos que aparecen a la derecha en todas las dependencias en las que aparecen: ninguno

Establecidos los tipos, podemos aplicar el algoritmo de extracción de claves candidatas:

1º Eliminación de atributos independientes:

No existen atributos independientes por lo que la relación sin independientes queda igual que la relación original R_{SI} = R = {A, B, C, D}

2º Eliminación de equivalencias:

Al existir una equivalencia entre B y C, debemos seleccionar un atributo para quedarse y otro para ser sustituido (elegimos C para que ser sustituido puesto que B aparece en un número mayor de dependencias funcionales). Por ello R_{SIE}= R_{SI}= R = {A, B, D} y F_{SIE} = {A→B, B→B, BD→A, A→D, B→B} = {A→B, BD→A, A→D}

Al eliminar dependencias, puede haber cambiado el comportamiento de algún atributo (independiente, sólo a la derecha, sólo a la izquierda, …) en el conjunto F_{SIE} pero se aprecia que no es el caso.

3º Procesamiento de atributos que aparecen sólo a la izquierda:

K_p = Æ

Dado que no hay candidato, es necesario proceder al paso 4.

4º Procesamiento de atributos que aparecen a la izquierda y a la derecha:

Para ello, tomamos el candidato del paso 3 (el conjunto vacío) y lo combinamos con todos los atributos que estén a la izquierda en alguna dependencia y a la derecha en otra y no hayan sido generado por el candidato del paso 3 (el conjunto vacío no puede generar nada). Es decir, habría que combinar el conjunto vacío (del paso 3) con los atributos {A, B y D}, puesto que no son generados por el vacío.

K_p'= {A, B, D}

Al haber candidatos, comprobamos si el primero es clave, calculando el conjunto de atributos incluidos en el cierre transitivo:

A^+ = {A, B, D} = R_{SIE} por lo que A es una clave para R_{SIE}, CK_{SIE} = {A} y K_p'= {B, D}

Procedemos con el candidato B:

B^+ = {B} $\neq$ R_{SIE} por lo que B no es una clave pero puede participar en alguna clave (al ser un atributo que está a la izquierda de alguna dependencia y a la derecha de otras). Para generar los candidatos que extienden a B, lo combinamos con otros atributos que estén a la izquierda de alguna dependencia funcional y a la derecha de otras, pero que no sea generado por el propio B (es decir, no esté incluido en B^+) y que no sean clave, lo que sólo nos deja D.

Podríamos incluir BD en el conjunto de candidatos K_p' pero, al estar incluido D, si descubrimos que D es una clave, BD quedaría invalidado como candidato (por ser una extensión de una clave). Si, por el contrario, no incluimos BD en K_p' y D resulta no ser clave, BD volverá a ser generado por combinación de D (si es que B no es generado en el conjunto D^+). Por ello, este paso no incluye un candidato al conjunto K_p' pero elimina a B como posible candidato, de modo que K_p'= {D}

Procedemos con el candidato D:

D^+ = {D} $\neq$ R_{SIE} por lo que D no es una clave pero puede participar en alguna clave (al ser un atributo que está a la izquierda de alguna dependencia y a la derecha de otras). Combinamos D con otros atributos que estén a la izquierda de alguna dependencia funcional y a la derecha de otras, pero que no sea generado por el propio D (es decir, no esté incluido en D^+) y que no sean clave, lo que nos deja con la combinación BD. Por tanto, K_p'= {BD}

Procedemos con el candidato BD:

BD^+ = {A, B, D} = R_{SIE} por lo que A es una clave para R_{SIE}, CK_{SIE} = {A, BD} y K_p'= Æ

Al no quedar candidatos que explorar, damos por concluido el paso 4.

5º Incorporación de atributos independientes:

Al no existir atributos independientes en R, este paso no modifica las claves obtenidas para la relación R_{SI} por lo que CK' = {A, BD}.

6º Incorporación de atributos equivalentes:

Al existir una equivalencia entre B y C en F, elegimos C para ser sustituido y B para quedarse. Puesto que B y C son equivalentes, si B participa en una clave de R, C debe participar en otra por lo que CK = {A, BD, CD}

Por tanto, podemos decir que A y CD son claves pero no las únicas. BD también es clave de R.

Enunciado

Sea R una relación de esquema {A, B, C, D, E}, con una única clave candidata E y que verifica un conjunto F de dependencias funcionales {E→C, C→A, C→D, E→B}. Comprobar si la relación está en Forma Normal de Boyce y Codd (sin comprobar si está en segunda forma normal o en tercera forma normal) y si no lo está, realizar una descomposición hasta que todas las relaciones que integran la descomposición estén en dicha forma normal.

Resolución

La relación R esté en Forma Normal de Boyce y Codd si, y sólo si, todas las dependencias tienen a su izquierda una clave candidata. Según esta norma, podemos decir que la relación no está en BCNF porque C→A y C→D forman parte de F y C no es clave candidata de la relación.

Para conseguir una normalización, debemos aplicar el Teorema de Heath sobre una de las dependencias que no cumplen la forma normal.

Empezamos normalizando por C→A:

R_1 = {C, A}, F_1 = {C→A}, CK_1 = {C}, **R_1 está en FNBC** puesto que todas las dependencias funcionales de F_1 tienen una clave candidata de R_1 a su izquierda.

R_2 = {B, C, D, E}, F_2 = {E→C, C→D, E→B}, CK_2 = {E}, R_2 no está en BCNF porque C→D forma parte de F_2 pero C no es clave candidata de R_2.

Es necesario normalizar la relación R_2 mediante la aplicación del Teorema de Heath aplicado sobre C→D:

$R_{2,1}$ = {C, D}, F_2 = {C→D}, CK_2 = {C}, **$R_{2,1}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,1}$.

$R_{2,2}$ = {B, C, E}, F_2 = {E→C, E→B}, CK_2 = {E}, **$R_{2,2}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,2}$.

El resultado de nuestra normalización es el conjunto de relaciones:

$$\{(\{C, A\}, r_1), (\{C, D\}, r_{2,1}), (\{B, C, E\}, r_{2,2})\}$$