Teórico 6

Diseño - Dependencias Funcionales

Problemas de un Mal Diseño

- Repetición de información (potencial inconsistencia).
- Imposibilidad de representar información.

Ejemplo

Supongamos que estamos trabajando con información de Personas y Vehículos, y dicha información está representada con una sola tabla.

R(<u>DNI</u>, NYApellido, Dir, <u>Npat</u>, Modelo, Marca)

Supongamos la siguiente instancia de la relación R:

DNI	NYApellic				
23	Juan Pere				
23	Juan Pere				
24	Carlos Pui				

Qué problemas se pueden apreciar ?

Problemas del Diseño Anterior

- Las personas que no tienen ningún vehículo asociado no las puedo representar debido a que la clave primaria no admite valores nulos.
- Se repite la información de las personas que tienen más de un vehículo.

Que Hacer ?

Las tablas que están mal diseñadas hay que dividirlas en dos o más tablas.

Una posibilidad puede ser dividir R en Personas y Vehículos de la siguiente forma:

Personas (DNI, NYApellido,Dir, Modelo)

Y

Vehiculos (Npat, Modelo,Marca)

Las Instancias Serían

Personas

DNI	NYApellido		
23	Juan Peres		
23	Juan Peres		
24	Carlos Puig		

Vehículos

Npat	Modelo	
NDT 454	2000	
KJI 566	2005	
GTR 654	2000	

Para que una división sea válida y no tenga pérdida de información, el join Natural entre las tablas de ser igual a la tabla original en cuanto a instancias.

r = personas $\bowtie$ vehículos

En nuestro caso: personas $\bowtie$ vehículos $\neq$ r

DNI	NYApellic				
23	Juan Pere				
23	Juan Pere				
23	Juan Pere				

Por lo Tanto

- Debo buscar un método que me permita hacer una buena división.

- Como determino si llegue a un buen diseño?

Dependencias Funcionales

- Son restricciones a los datos que puede admitir una base de datos.
- El concepto de dependencia funcional es una generalización del concepto de Clave.
- Las dependencias funcionales juegan un papel importante en el diseño de bases de datos.

Superclave

Definición:

Sea R un esquema de relación, se dice que un subconjunto K de R es una superclave de R si para todo par de tuplas $t1, t2 \in r(R)$ tales que $t1 \neq t2$ entonces $t1[k] \neq t2[k]$

o de manera equivalente,

$t1[k] = t2[k]$ entonces $t1 = t2$

Dependencia Funcional

Definición:

- Sea R un esquema de relación y sean $\alpha \subseteq R$ y $\beta \subseteq R$, la dependencia funcional

  $\alpha \rightarrow \beta$ (se lee "α determina β")

  se cumple en R, si para cualquier instancia r(R) se cumple que para todo par de tuplas $t_1, t_2 \in r$, si $t_1[\alpha] = t_2[\alpha]$ entonces $t_1[\beta] = t_2[\beta]$

Ejemplo

1) Dada R(A,B) con instancia r(R)

A	B
1	2
2	2
3	3

Por ej. Vale A → B y no Vale B → A

2) Dada R(A,B,C) con instancia r(R)

A	B	C
1	1	1
2	2	2
3	2	2
2	2	3

Por ej. Vale A → B y C → B y no Vale C → A

Definiciones

- Sea $K \subseteq R$, K es una superclave para el esquema R si y sólo si: $K \rightarrow R$.
- Sea $K \subseteq R$, K es una clave candidata para R si y sólo si:
  - $K \rightarrow R$, y
  - No existe un $\alpha \subset K$ *tal que* $\alpha \rightarrow R$
- Las dependencias funcionales permiten expresar restricciones que no pueden ser expresadas mediante superclaves.

Dependencia Funcional (Cont.)

- Una dependencia funcional es trivial si es satisfecha por todas la instancias de una relación.
  - E.j.
    - DNI, NYApellido $\rightarrow$ DNI
    - NYApellido $\rightarrow$ NYApellido
  - En general, $\alpha \rightarrow \beta$ es trivial si $\beta \subseteq \alpha$

Clausura de un Conjunto de Dependencias Funcionales

- A partir de un conjunto de dependencias funcionales F se puede probar que valen otras. Se dice que estas dependencia funcionales están implicadas lógicamente por F.

Ejemplo

Dado un esquema R=(A,B,C) donde valen las dependencia funcionales $A \rightarrow B$ y $B \rightarrow C$, se puede probar que $A \rightarrow C$ también vale:

Para todo par de tuplas t_1, t_2 de r, tales que $t_1[A] = t_2[A]$

Dado que vale $A \rightarrow B$ entonces se cumple que

$t_1[B] = t_2[B]$

Además como vale $B \rightarrow C$ se cumple que

$t_1[C] = t_2[C]$

De esta forma queda probada la validez de $A \rightarrow C$

Definición de F^+

Sea F un conjunto de dependencias funcionales que valen en un esquema R, la clausura de F, denotado F^+, es el conjunto de dependencias funcionales implicadas por F.

Axiomas de Amstrong

- Para simplificar el cálculo de F^+, evitando la utilización de la definición de dependencia funcional para obtener nuevas dependencias, se pueden utilizar un conjunto de reglas llamadas Axiomas de Amstrong.

Axiomas de Amstrong(Cont)

Para las siguientes reglas se utilizan letras griegas(α,β,γ, etc) para notar conjuntos de atributos y se utiliza $\alpha\beta$ para notar $\alpha \cup \beta$:

- **Reflexividad**
  si $\beta \subseteq \alpha$, entonces $\alpha \rightarrow \beta$

- **Aumentación**
  si $\alpha \rightarrow \beta$ *y* γ es un conjunto de atributos , entonces $\gamma\alpha \rightarrow \gamma\beta$

- **Transitividad**
  si $\alpha \rightarrow \beta$, *y* $\beta \rightarrow \gamma$, entonces $\alpha \rightarrow \gamma$

Estas reglas son correctas porque no generan dependencias incorrectas y son completas porque para un F dado permiten generar todo F^+.

Axiomas de Amstrong(Cont)

- Para simplificar más el cálculo de F^+ se ofrecen más reglas que pueden ser probadas utilizando los Axiomas de Amstrong:

  - **Unión**
    Si se cumple $\alpha \rightarrow \beta$ *y* $\alpha \rightarrow \gamma$, entonces se cumple $\alpha \rightarrow \beta\gamma$

  - **Descomposición**
    Si se cumple $\alpha \rightarrow \beta\gamma$, entonces se cumple $\alpha \rightarrow \beta$ *y* $\alpha \rightarrow \gamma$

  - **Seudotransitividad**
    Si se cumple $\alpha \rightarrow \beta$ *y* $\gamma\beta \rightarrow \delta$, entonces $\alpha\gamma \rightarrow \delta$

Ejemplo

Dado R=(A,B,C,D) F={$A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$}

Algunas dependencias de F^+ son:

- Por transitividad de $A \rightarrow B$ y $B \rightarrow C$ tenemos:

$A \rightarrow C$

- Por aumentación de $A \rightarrow B$ con D tenemos:

$AD \rightarrow BD$

Clausura de Conjuntos de Atributos

- Dado un conjunto de atributos α, se define la clausura de α bajo *F* (denotado por α^+) como el conjunto de atributos que son determinados funcionalmente por α bajo *F:*

$$\alpha \rightarrow \beta \textit{ está } \text{en } F^+ \Longleftrightarrow \beta \subseteq \alpha^+$$

- Algoritmo para computar α^+ bajo F

resultado := α;
while (cambia resultado) **do**
  for each dependencia $\beta \rightarrow \gamma$ **in** *F* **do**
    begin
    if $\beta \subseteq$ *resultado*
      then *resultado* := *resultado* $\cup\ \gamma$
    end

Ejemplo de Clausura de Atributos

- *Dados* $R = (A, B, C, G, H, I)$ *y*

  $F = \{A \rightarrow B$

  $A \rightarrow C$

  $CG \rightarrow H$

  $CG \rightarrow I$

  $B \rightarrow H\}$

- Calcular $(AG)^+$
  1. *resultado* $= AG$
  2. *resultado* $= ABCG$ $(A \rightarrow C$ *y* $A \rightarrow B)$
  3. *resultado* $= ABCGH$ $(CG \rightarrow H$ y $CG \subseteq AGBC)$
  4. *resultado* $= ABCGHI$ $(CG \rightarrow I$ y $CG \subseteq AGBCH)$

- AG es una clave candidata?
  1. AG es superclave?
     1. Vale $AG \rightarrow R$?
  1. Algún subconjunto de AG es superclave?
     1. vale $A^+ \rightarrow R$?
     2. vale $G^+ \rightarrow R$?

Uso de la Clausura de Conjunto de Atributos

El algoritmo de clausura de un conjunto de atributos tiene varios usos:

- Prueba para superclaves:
  - Para probar si α es una superclave, computar α^+ y chequear si α^+ contiene todos los atributos de *R*.

- Probar la validez de una dependencia funcional:
  - Para chequear si una dependencia funcional $\alpha \rightarrow \beta$ vale, debe chequearse si $\beta \subseteq \alpha^+$.
  - Esto es computar α^+ utilizando la clausura de atributos, y luego chequear si la clausura contiene a β.

- Computar la clausura de F:
  - Para cada $\gamma \subseteq R$, definir la clausura γ^+, y para cada $S \subseteq \gamma^+$, generar la dependencia funcional $\gamma \rightarrow S$.

Recubrimiento Canónico

- Los conjuntos de dependencias funcionales pueden tener dependencias redundantes que se pueden deducir de las otras:
  - Ej: $A \rightarrow C$ es redundante en: $\{A \rightarrow B,\ B \rightarrow C,\ A \rightarrow C\}$
  - Parte de una dependencia funcional puede ser redundante
    - Ej. de redundancia en la parte derecha

      $\{A \rightarrow B,\ B \rightarrow C,\ A \rightarrow CD\}$ puede ser simplificado a $\{A \rightarrow B,\ B \rightarrow C,\ A \rightarrow D\}$
    - Ej. de redundancia en la parte izquierda :

      $\{A \rightarrow B,\ B \rightarrow C,\ AC \rightarrow D\}$ puede ser reducido a $\{A \rightarrow B,\ B \rightarrow C,\ A \rightarrow D\}$

- Intuitivamente, un recubrimiento canónico de F es un conjunto "mínimo" de dependencias funcionales equivalente a F, sin dependencias o partes de dependencias redundantes.

Atributos Extraños

- Considere el conjunto de dependencias funcionales F y una dependencia funcional $\alpha \rightarrow \beta$ en F:
  - El Atributo A es **extraño** en α si $A \in \alpha$ y F implica lógicamente a $(F - \{\alpha \rightarrow \beta\}) \cup \{(\alpha - A) \rightarrow \beta\}$.
  - El Atributo A es **extraño** en β si $A \in \beta$ y el conjunto de dependencias $(F - \{\alpha \rightarrow \beta\}) \cup \{\alpha \rightarrow (\beta - A)\}$ implica lógicamente a F.

- Ej.:: Dado $F = \{A \rightarrow C, B \rightarrow C, AC \rightarrow D\}$
  - C es extraño en $AC \rightarrow D$ porque F implica lógicamente a $A \rightarrow C, B \rightarrow C, A \rightarrow D$.

    Se prueba aumentando con A y transitividad
- Ej.: dado $F = \{A \rightarrow C, AB \rightarrow CD\}$
  - C es extraño en $AB \rightarrow CD$ porque $\{A \rightarrow C, AB \rightarrow D\}$ implica lógicamente a F.

    se prueba con la regla de Aumentación, Unión y Descomposición...

Prueba si un Atributo es Raro

- Considere un conjunto *F* de dependencias funcionales y la dependencia $\alpha \rightarrow \beta$ in *F.*
- Para probar si el Atributo $A \in \alpha$ es extraño en α
  1. computar $(\{\alpha\} - A)^+$ utilizando las dependencias en *F.*
  2. chequear que $(\{\alpha\} - A)^+$ contiene a β; si es así, *A* es extraño.
- Para probar si el atributo $A \in \beta$ es extraño en β
  1. computar α^+ utilizando sólo las dependencias en
     $F' = (F - \{\alpha \rightarrow \beta\}) \cup \{\alpha \rightarrow (\beta - A)\}$,
  2. chequear que α^+ contiene a *A; si es así, A* es extraño

Recubrimiento Canónico

- Un ***recubrimiento canónica*** para F es un conjunto de dependencias F_c *,* tal que:
  - F implica lógicamente todas las dependencias en $F_{c,}$ y
  - F_c implica lógicamente todas las dependencias en F, y
  - Ninguna dependencia en F_c contiene atributos extraños, y
  - Cada lado izquierdo de las dependencias en F_c es único.
- Para computar un recubrimiento canónico para F:
  repeat
  Usar la regla de unión para reemplazar las dependencias in F
  $\alpha_1 \rightarrow \beta_1$ y $\alpha_1 \rightarrow \beta_2$ con $\alpha_1 \rightarrow \beta_1 \beta_2$
  Encontrar una dependencia funcional $\alpha \rightarrow \beta$ con un
  atributo extraño en α o en β
  Si se encuentra algún atributo extraño, eliminarlo de $\alpha \rightarrow \beta$
  until F no cambie

Nota: La regla de Unión debería ser aplicada después de que algunos atributos extraños han sido eliminados.

Ejemplo

- Dado $R = (A, B, C)$ y
  $F = \{A \rightarrow BC$
  $B \rightarrow C$
  $A \rightarrow B$
  $AB \rightarrow C\}$
- Combinar $A \rightarrow BC$ y $A \rightarrow B$ *en* $A \rightarrow BC$
  - El conjunto queda $\{A \rightarrow BC, B \rightarrow C, AB \rightarrow C\}$
- A es extraño en $AB \rightarrow C$ porque $B \rightarrow C$ implica lógicamente $AB \rightarrow C$.
  - Luego el conjunto es $\{A \rightarrow BC, B \rightarrow C\}$
- C es extraño en $A \rightarrow BC$ porque $A \rightarrow BC$ es implicado lógicamente por $A \rightarrow B$ ***y*** $B \rightarrow$ C.
- El recubrimiento canónico es:

$$A \rightarrow B$$
$$B \rightarrow C$$