

Bases de Datos 1

Alejandra Lliteras

alejandra.lliteras@lifa.info.unlp.edu.ar



En la clase anterior...

Teoría de diseño para bases de datos relaciones

Dependencia Multivaluada



Teoría de diseño para bases de datos relaciones

Dependencia Multivaluada



Teoría de diseño de BBDD relacionales

► Dependencia Multivaluada

- Una dependencia multivaluada, afirma que dos o mas atributos son independientes del resto
- Como consecuencia de la independencia, se tiene redundancia. Esta redundancia no se elimina con las dependencias funcionales

Teoría de diseño de BBDD relacionales

► Dependencia Multivaluada

- se puede decir que: $X \twoheadrightarrow Y$ si dado un valor de X , hay un conjunto de valores de Y asociados y este conjunto de valores de Y **NO** está relacionado (ni funcional ni multifuncionalmente) con los valores de $R - X - Y$ (donde R es el esquema), es decir Y es independiente de los atributos de $R - X - Y$.

Teoría de diseño de BBDD relacionales

► Dependencia Multivaluada

(otra forma de definirla)

Sea R un esquema de relación

La Dependencia Multivaluada $X \twoheadrightarrow Y$ vale en R si \forall los pares de tuplas t_1 y t_2 en R , tal que

- $t_1[X] = t_2[X]$ existen las tuplas t_3 y t_4 en R tales que:
 - $t_1[X] = t_2[X] = t_3[X] = t_4[X]$
 - $t_3[Y] = t_1[Y]$
 - $t_3[R-X-Y] = t_2[R-X-Y]$
 - $t_4[Y] = t_2[Y]$
 - $t_4[R-X-Y] = t_1[R-X-Y]$

Teoría de diseño para bases de datos relaciones

Dependencia Multivaluada trivial



Teoría de diseño de BBDD relacionales

► Dependencia Multivaluada trivial

Sea R un esquema de relación

Una dependencia multivaluada de la forma $X \twoheadrightarrow Y$ que vale en R es trivial si:

- el conjunto de atributos X, Y son todos los atributos del esquema

Teoría de diseño de BBDD relacionales

¿ Cómo proceder cuando se hallan dependencias multivaluadas?

Teoría de diseño de bases de datos relacionales

Formas normales
4FN

Teoría de diseño de BBDD relacionales

- ▶ Cuarta Forma Normal (4FN)
 - Un esquema R está en 4NF con respecto a un conjunto de dependencias multivaluadas D , si \forall dependencia multivaluada de la forma $X \twoheadrightarrow Y$ se cumple que:
 - $X \twoheadrightarrow Y$ es una dependencia multivaluada trivial

Teoría de diseño de BBDD relacionales

► Cuarta Forma Normal (4FN)

En otras palabras:

- Un esquema está en 4FN cuando:
 - No tiene dependencias multivaluadas
- O bien,
- Las dependencias multivaluadas que en él valen, son triviales.

En esta clase...

Teoría de diseño para bases de datos relaciones

Dependencias Multivaluadas Ejemplos



Teoría de diseño de BBDD relacionales

Supongamos que se tiene el siguiente esquema

HOSPITALES(cod_hospital, medico, enfermero, administrativo)

Donde:

Cada Hospital tiene un código único

Un médico puede trabajar en varios hospitales

Un enfermero puede trabajar en varios hospitales

Un administrativo puede trabajar en varios hospitales

| cod_hospital | medico | enfermero | administrativo |
|--------------|--------|-----------|----------------|
| h1 | m1 | e1 | a1 |
| h1 | m1 | e2 | a1 |
| h1 | m1 | e1 | a2 |
| h1 | m1 | e2 | a2 |
| h1 | m2 | e1 | a1 |
| h1 | m2 | e2 | a1 |
| h1 | m2 | e1 | a2 |
| h1 | m2 | e2 | a2 |

Teoría de diseño de BBDD relacionales

HOSPITALES(cod_hospital, medico, enfermero, administrativo)

Donde:

Cada Hospital tiene un código único

Un médico puede trabajar en varios hospitales

Un enfermero puede trabajar en varios hospitales

Un administrativo puede trabajar en varios hospitales

Por las restricciones mencionadas, los médicos no están relacionados con los enfermeros, ni con los administrativos. Así como tampoco los enfermeros con los administrativos ni con los médicos (idem con los administrativos con el resto de los atributos)

DM1) cod_hospital --->> medico

DM2) cod_hospital --->> enfermero

DM3) cod_hospital --->> administrativo

En particular, ¿qué implica decir:

$\text{cod_hospital} \twoheadrightarrow \text{medico}$

En términos de una de las definiciones vistas para el concepto de dependencia funcional?

Sea R un esquema de relación, la Dependencia Multivaluada $X \twoheadrightarrow Y$ vale en R si \forall los pares de tuplas t_1 y t_2 en R , tal que $t_1[X] = t_2[X]$ existen las tuplas t_3 y t_4 en R tales que:

$$t_1[X] = t_2[X] = t_3[X] = t_4[X]$$

$$t_3[Y] = t_1[Y]$$

$$t_3[R-X-Y] = t_2[R-X-Y]$$

$$t_4[Y] = t_2[Y]$$

$$t_4[R-X-Y] = t_1[R-X-Y]$$

HOSPITALES(cod_hospital,medico, enfermero, administrativo)

Si decimos que vale la DM $\text{cod_hospital} \twoheadrightarrow \text{medico}$, seguro tienen que existir las tuplas t_1, t_2, t_3 y t_4 que menciona la definición

$$t_1[X] = t_2[X] = t_3[X] = t_4[X] = h1$$

$$t_1[X] = t_2[X] = t_3[X] = t_4[X] = h1$$

$$t_3[Y] = t_1[Y]$$

$$t_3[R-X-Y] = t_2[R-X-Y]$$

$$t_4[Y] = t_2[Y]$$

$$t_4[R-X-Y] = t_1[R-X-Y]$$

| | cod_hospital | medico | enfermero | administrativo |
|-----------|---------------------|---------------|------------------|-----------------------|
| | h1 | m1 | e1 | a1 |
| t1 | h1 | m1 | e2 | a1 |
| | h1 | m1 | e1 | a2 |
| | h1 | m1 | e2 | a2 |
| | h1 | m2 | e1 | a1 |
| | h1 | m2 | e2 | a1 |
| t2 | h1 | m2 | e1 | a2 |
| | h1 | m2 | e2 | a2 |

$$t_3[\text{medico}] = t_1[\text{medico}] = m1$$

$$t3 = (h1, m1, e1, a2)$$

$$t_3[\text{enfermero,administrativo}] = t_2[\text{enfermero,administrativo}] = e1,a2$$

$$t_4[\text{medico}] = t_2[\text{medico}] = m2$$

$$t4 = (h1, m2, e2, a1)$$

$$t_4[\text{enfermero,administrativo}] = t_1[\text{enfermero,administrativo}] = e2,a1$$

Teoría de diseño de BBDD relacionales

HOSPITALES(cod_hospital,medico, enfermero, administrativo)

cod_hospital -->> medico

$$t_1[X] = t_2[X] = t_3[X] = t_4[X] = h1$$

$$t_1[X] = t_2[X] = t_3[X] = t_4[X]$$

$$t_3[Y] = t_1[Y]$$

$$t_3[R-X-Y] = t_2[R-X-Y]$$

$$t_4[Y] = t_2[Y]$$

$$t_4[R-X-Y] = t_1[R-X-Y]$$

| | cod_hospital | medico | enfermero | administrativo | |
|-----------|--------------|--------|-----------|----------------|------|
| | h1 | m1 | e1 | a1 | |
| t1 | h1 | m1 | e2 | a1 | |
| | h1 | m1 | e1 | a2 | → t3 |
| | h1 | m1 | e2 | a2 | |
| | h1 | m2 | e1 | a1 | |
| | h1 | m2 | e2 | a1 | → t4 |
| t2 | h1 | m2 | e1 | a2 | |
| | h1 | m2 | e2 | a2 | |

$$t_3[\text{medico}] = t_1[\text{medico}] = m1$$

$$t3 = (h1, m1, e1, a2)$$

$$t_3[\text{enfermero,administrativo}] = t_2[\text{enfermero,administrativo}] = e1,a2$$

$$t_4[\text{medico}] = t_2[\text{medico}] = m2$$

$$t4 = (h1, m2, e2, a1)$$

$$t_4[\text{enfermero,administrativo}] = t_1[\text{enfermero,administrativo}] = e2,a1$$

Teoría de diseño de BBDD relacionales

Dependencia Multivaluada trivial



Teoría de diseño de BBDD relacionales

► Dependencia Multivaluada trivial

Sea R un esquema de relación

Una dependencia multivaluada de la forma $X \twoheadrightarrow Y$ que vale en R es trivial si:

- el conjunto de atributos X, Y son todos los atributos del esquema
- o bien,
- $Y \subseteq X$

Teoría de diseño de BBDD relacionales

► Retomemos el ejercicio:

ATENCIONES(codHospital, nombreHospital, dniPaciente, legajoPaciente, dniMedico)

Donde:

- Un paciente tiene asignado para cada hospital un número de legajo
- Un legajo en un hospital se asigna a una única persona
- En un hospital trabajan muchos médicos y un médico puede trabajar en diversos hospitales
- Un médico atiende a muchos pacientes
- Cada hospital posee un nombre y el mismo nombre se puede repetir para diferentes hospitales
- Un paciente se atiende en muchos hospitales y de cada hospital que se atiende se registran los médicos que lo atienden

Para este ejercicio, la clase pasada hallamos las siguientes dependencias funcionales y claves candidatas:

ATENCIONES(codHospital, nombreHospital, dniPaciente, legajoPaciente, dniMedico)

Dependencias Funcionales

df1) codHospital → nombreHospital

df2) legajoPaciente, codHospital → dniPaciente

df3) dniPaciente, codHospital → legajoPaciente

Claves Candidatas

cc1) {codHospital, legajoPaciente, dniMedico}

cc2) {codHospital, dniPaciente, dniMedico}

En base a las dependencias funcionales y claves candidatas, aplicamos el proceso de normalización hasta BCNF y llegamos a las siguientes particiones en dicha forma normal.

A1 (codHospital, nombreHospital)

A3 (codHospital, legajoPaciente, dniPaciente)

A4 (codHospital, legajoPaciente, dniMedico)

ATENCIONES(codHospital, nombreHospital, dniPaciente, legajoPaciente, dniMedico)

A4 (codHospital, legajoPaciente, dniMedico)

En A4, vimos que valía la dependencia multivaluada:

DM1) codHospital, legajoPaciente \twoheadrightarrow dniMedico

Y vimos la definición de Dependencia Multivaluada trivial

Sea R un esquema de relación

Una dependencia multivaluada de la forma $X \twoheadrightarrow Y$ que vale en R es trivial si:

- el conjunto de atributos X,Y son todos los atributos del esquema
- bien,
- $Y \subseteq X$

Entonces, por la definición anterior, la DM1 es trivial en A4

Teoría de diseño de BBDD relacionales

Dependencia Multivaluada
Caso especial

\emptyset (vacío)

Teoría de diseño de BBDD relacionales

► Caso especial de dependencia multivaluada

Supongamos la relación:

CasaAlquilerVideo (#video, titulo, #cliente, nyAp_cliente)

Donde

- #Cliente representa a los clientes que alguna vez realizaron un alquiler
- #video son todos los videos que hay en la casa de alquiler

Dep. funcionales:

df1: #video → titulo

df2: #cliente → nyAp_cliente

Clave candidata: {#cliente, #video}

Teoría de diseño de BBDD relacionales

► Caso especial de dependencia multivaluada

Supongamos la relación:

CasaAlquilerVideo (#video, titulo, #cliente, nyAp_cliente)

Dep. funcionales:

df1: #video → titulo

df2: #cliente → nyAp_cliente

Clave candidata: {#cliente, #video}

Notamos que CasaAlquilerVideo no cumple con la definición de BCNF ya que existe, por ejemplo la df1, donde {#video} no es superclave del esquema. Entonces se particiona CasaAlquilerVideo, considerando la df1

Teoría de diseño de BBDD relacionales

► Caso especial de dependencia multivaluada

Supongamos la relación:

CasaAlquilerVideo (#video, titulo, #cliente, nyAp_cliente)

Dep. funcionales:

df1: #video → titulo

df2: #cliente → nyAp_cliente

Clave candidata: {#cliente, #video}

Notamos que CasaAlquilerVideo no cumple con la propiedad de BCNF ya que existe, por ejemplo la df1, donde {#video} no es superclave del esquema. Entonces se particiona CasaAlquilerVideo, considerando la df1:

R1 (#video, titulo)

R2 (#video, #cliente, nyAp_cliente)

Teoría de diseño de BBDD relacionales

► Caso especial de dependencia multivaluada

Supongamos la relación:

CasaAlquilerVideo (#video, titulo, #cliente, nyAp_cliente)

Dep. funcionales:

df1: #video → titulo

df2: #cliente → nyAp_cliente

Clave candidata: {#cliente, #video}

R1 (#video, titulo)

R2 (#video, #cliente, nyAp_cliente)

- R1 cumple con BCNF ya que la única df que vale es df1 y {#video} es superclave en R1
- R2 no cumple con BCNF por la existencia de la df2, cuyo determinante no es superclave en R2, particiono

Teoría de diseño de BBDD relacionales

► Caso especial de dependencia multivaluada

Supongamos la relación:

CasaAlquilerVideo (#video, titulo, #cliente, nyAp_cliente)

Dep. funcionales:

df1: #video → titulo

df2: #cliente → nyAp_cliente

Clave candidata: {#cliente, #video}

R2 (#video, #cliente, nyAp_cliente)

Particionamos el esquema R2 contemplando la df2 (por lo antes descripto)

R3 (#cliente, nyAp_cliente)

R4 (#video, #cliente)

Teoría de diseño de BBDD relacionales

► Caso especial de dependencia multivaluada

Supongamos la relación:

CasaAlquilerVideo (#video, titulo, #cliente, nyAp_cliente)

Dep. funcionales:

df1: #video → titulo

df2: #cliente → nyAp_cliente

Clave candidata: {#cliente, #video}

R3 (#cliente, nyAp_cliente)

R4 (#video, #cliente)

- R3 cumple con BCNF ya que la única df que vale es df2 y {#cliente} es superclave en R3
- R4 cumple con BCNF, todos sus atributos forman parte de la clave

Teoría de diseño de BBDD relacionales

- ▶ Caso especial de dependencia multivaluada

R4 (#video, #cliente)

Donde

- #Cliente representa a los clientes que alguna vez realizaron un alquiler
- #video son todos los videos que hay en la casa de alquiler

Supongamos un ejemplo con datos particulares en la partición R4.
Que forma tendrían que tener?

| <u>#cliente</u> | <u>#video</u> |
|-----------------|---------------|
| c1 | v1 |
| c1 | v2 |
| c1 | v3 |
| c2 | v1 |
| c2 | v2 |
| c2 | v3 |

Teoría de diseño de BBDD relacionales

► Caso especial de dependencia multivaluada

R4 (#video, #cliente)

Donde

- #Cliente representa a los clientes que alguna vez realizaron un alquiler
- #video son todos los videos que hay en la casa de alquiler

► Se dice en este caso, que en R4 valen las siguientes dependencias multivaluadas:

- DM1) $\emptyset \twoheadrightarrow \#cliente$
- DM2) $\emptyset \twoheadrightarrow \#video$

Teoría de diseño de BBDD relacionales

¿ Cómo proceder cuando se hallan dependencias multivaluadas?

Teoría de diseño de BBDD relacionales

Formas normales
4FN

Teoría de diseño de BBDD relacionales

- ▶ 4FN (Cuarta Forma Normal)
 - Un esquema R está en 4NF con respecto a un conjunto de dependencias multivaluadas D , si \forall dependencia multivaluada de la forma $X \twoheadrightarrow Y$ se cumple que:
 - $X \twoheadrightarrow Y$ es una dependencia multivaluada trivial

Teoría de diseño de BBDD relacionales

► 4FN (Cuarta Forma Normal)

En otras palabras:

- Un esquema está en 4FN cuando:
 - No tiene dependencias multivaluadas
- O bien,
- Las dependencias multivaluadas que en él valen, son triviales.

Teoría de diseño de BBDD relacionales

Habíamos estado trabajando sobre el esquema FIESTAS

FIESTAS (#salon, dirección, capacidad, fecha_fiesta, nom_contratante, cant_invitados, nombre_invitado, cant_mesas, mesa_invitado, servicio_contratado, dir_contratante, dni_invitado)

- En cada salón se realiza una sola fiesta por día, en un día puede haber varias fiestas en diferentes salones.
- Para cada fiesta puede figurar más de un contratante
- Cada invitado a una fiesta tiene asociado un número de mesa
- La cantidad de mesas del salón varia para cada fiesta
- Servicio contratado es una lista que describe los tipos de comida contratados para una fiesta (Esto quiere decir que para cada comida de la fiesta, aparece una tupla diferente en el esquema)
- Una persona puede ir a más de una fiesta en el mismo salón en diferentes fechas
- Una persona puede ir a mas de una fiesta el mismo día

Teoría de diseño de BBDD relacionales

FIESTAS (#salon, dirección, capacidad, fecha_fiesta, nom_contratante, cant_invitados, nombre_invitado, cant_mesas, mesa_invitado, servicio_contratado, dir_contratante, dni_invitado)

Y habíamos hallado las dependencias funcionales y la clave candidata válidas en el esquema FIESTAS:

Dependencias Funcionales

1. #salon → dirección, capacidad
2. nom_contratante → dir_contratante
3. #salon, fecha_fiesta, dni_invitado → mesa_invitado
4. #salon, fecha_fiesta → cant_invitados, cant_mesas
5. dni_invitado → nombre_invitado

Clave candidata: (#salon, fecha_fiesta, dni_invitado, nom_contratante, servicio_contratado)

Teoría de diseño de BBDD relacionales

FIESTAS (#salon, dirección, capacidad, fecha_fiesta, nom_contratante, cant_invitados, nombre_invitado, cant_mesas, mesa_invitado, servicio_contratado, dir_contratante, dni_invitado)

Y luego del proceso de particionamiento para llevar a BCNF, llegamos a las siguientes particiones:

F1(#salon, direccion, capacidad)

F3 (nom_contratante, dir_contratante)

F5(#salon, fecha_fiesta,dni_invitado, mesa_invitado)

F7(#salon, fecha_fiesta, cant_invitados, cant_mesas)

F9(dni_invitado, nombre_invitado)

F10(#salon, fecha_fiesta, nom_contratante, servicio_contratado, dni_invitado)

Teoría de diseño de BBDD relacionales

F10(#salon, fecha_fiesta, nom_contratante, servicio_contratado,
dni_invitado)

¿Tiene dependencias multivaluadas F10?

- En cada salón se realiza una sola fiesta por día, en un día puede haber varias fiestas en diferentes salones.
- Para cada fiesta puede figurar más de un contratante
- Cada invitado a una fiesta tiene asociado un número de mesa
- La cantidad de mesas del salón varia para cada fiesta
- Servicio contratado es una lista que describe los tipos de comida contratados para una fiesta (Esto quiere decir que para cada comida de la fiesta, aparece una tupla diferente en el esquema)
- Una persona puede ir a más de una fiesta en el mismo salón en diferentes fechas
- Una persona puede ir a mas de una fiesta el mismo día

Teoría de diseño de BBDD relacionales

F10(#salon, fecha_fiesta, nom_contratante, servicio_contratado, dni_invitado)

¿Las siguientes propuestas son válidas de acuerdo a la semántica de los atributos?

#salon -->> DNI_Invitado **X** ? El invitado depende de la fecha de la fiesta
#salon, fecha_fiesta -->> DNI_invitado ? **✓**

| #salón | Fecha_fiesta | Nom_contr | Serv_contr | Dni_invitado |
|--------|--------------|---------------|----------------|--------------|
| #1 | 10/03/12 | Pedro Guti | empanadas | 11111111 |
| #1 | 10/03/12 | Pedro Guti | empanadas | 11222222 |
| #1 | 10/03/12 | Pedro Guti | pizza | 11111111 |
| #1 | 10/03/12 | Pedro Guti | pizza | 11222222 |
| #1 | 10/03/12 | María Zeta | empanadas | 11111111 |
| #1 | 10/03/12 | María Zeta | empanadas | 11222222 |
| #1 | 10/03/12 | María Zeta | pizza | 11111111 |
| #1 | 10/03/12 | María Zeta | pizza | 11222222 |
| #1 | 11/03/12 | Juan Zeballos | Mesa de quesos | 11333333 |
| #1 | 11/03/12 | Juan Zeballos | calentitos | 11333333 |
| ... | ... | ... | ... | ... |

F10 (#salon, fecha_fiesta, nom_contratante, servicio_contratado, dni_invitado)

Al tomar la siguiente dependencia multivaluada como válida

#salon, fecha_fiesta \twoheadrightarrow DNI_invitado

Y de acuerdo con una de las definiciones de dependencia multivaluada que vimos, debe suceder que:

Sea R un esquema de relación, la Dependencia Multivaluada $X \twoheadrightarrow Y$ vale en R si \forall los pares de tuplas t_1 y t_2 en R, tal que $t_1[X] = t_2[X]$ existen las tuplas t_3 y t_4 en R tales que:

$$t_1[X] = t_2[X] = t_3[X] = t_4[X]$$

$$t_3[Y] = t_1[Y]$$

$$t_3[R-X-Y] = t_2[R-X-Y]$$

$$t_4[Y] = t_2[Y]$$

$$t_4[R-X-Y] = t_1[R-X-Y]$$

Veamos que forma deberían tener las tuplas t_1 , t_2 , t_3 y t_4 :

F10 (#salon, fecha_fiesta, nom_contratante, servicio_contratado, dni_invitado)

Tomemos dos tuplas t_1 y t_2 de F10, tal que

$t_1[X] = t_2[X]$, por ejemplo, tuplas que cumplan con $X = \{\text{\#salon, fecha_fiesta}\}$

| | #salón | Fecha_fiesta | Nom_contr | Serv_contr | Dni_invitado |
|------|--------|--------------|---------------|----------------|--------------|
| | #1 | 10/03/12 | Pedro Guti | empanadas | 11111111 |
| t1 → | #1 | 10/03/12 | Pedro Guti | empanadas | 11222222 |
| | #1 | 10/03/12 | Pedro Guti | pizza | 11111111 |
| | #1 | 10/03/12 | Pedro Guti | pizza | 11222222 |
| | #1 | 10/03/12 | María Zeta | empanadas | 11111111 |
| | #1 | 10/03/12 | María Zeta | empanadas | 11222222 |
| | #1 | 10/03/12 | María Zeta | pizza | 11111111 |
| t2 → | #1 | 10/03/12 | María Zeta | pizza | 11222222 |
| | #1 | 11/03/12 | Juan Zeballos | Mesa de quesos | 11333333 |
| | #1 | 11/03/12 | Juan Zeballos | calentitos | 11333333 |
| | ... | ... | ... | ... | ... |

F10 (#salon, fecha_fiesta, nom_contratante, servicio_contratado, dni_invitado)

Tomemos dos tuplas t_1 y t_2 de F10, tal que

$t_1[X] = t_2[X]$, por ejemplo, tuplas que cumplan con $X = \{\text{\#salon, fecha_fiesta}\}$

Sabiendo que \forall los pares de tuplas t_1 y t_2 en R, tal que

$t_1[X] = t_2[X]$ existen las tuplas t_3 y t_4 en R tales que:

$$t_1[X] = t_2[X] = t_3[X] = t_4[X]$$

$$t_3[Y] = t_1[Y]$$

$$t_3[R-X-Y] = t_2[R-X-Y]$$

$$t_4[Y] = t_2[Y]$$

$$t_4[R-X-Y] = t_1[R-X-Y]$$

¿Qué forma tendrían t_3 y t_4 si nuevamente, buscáramos un ejemplo con datos?

$$t_1[X] = t_2[X] = t_3[X] = t_4[X] \text{ con } X = \{\text{\#salon, fecha_fiesta}\}$$

#salon, fecha_fiesta --> > DNI_invitado

| | #salón | Fecha_fiesta | Nom_contr | Serv_contr | Dni_invitado |
|------|--------|--------------|---------------|----------------|---------------|
| t1 → | #1 | 10/03/12 | Pedro Guti | empanadas | 11111111 |
| | #1 | 10/03/12 | Pedro Guti | empanadas | 11222222 ← t4 |
| | #1 | 10/03/12 | Pedro Guti | pizza | 11111111 |
| | #1 | 10/03/12 | Pedro Guti | pizza | 11222222 |
| | #1 | 10/03/12 | María Zeta | empanadas | 11111111 |
| | #1 | 10/03/12 | María Zeta | empanadas | 11222222 |
| | #1 | 10/03/12 | María Zeta | pizza | 11111111 ← t3 |
| t2 → | #1 | 10/03/12 | María Zeta | pizza | 11222222 |
| | #1 | 11/03/12 | Juan Zeballos | Mesa de quesos | 11333333 |
| | #1 | 11/03/12 | Juan Zeballos | calentitos | 11333333 |
| | ... | ... | ... | ... | ... |

$$t_3[Y] = t_1[Y]$$

$$t_3[\text{dni_invitado}] = t_1[\text{dni_invitado}] = \{11111111\}$$

$$t_3[R-X-Y] = t_2[R-X-Y]$$

$$t_3[\text{nom_contr}, \text{servo_contr}] = t_2[\text{nom_contr}, \text{serv_contr}] = \{\text{María Zeta}, \text{pizza}\}$$


$$t_4[Y] = t_2[Y]$$

$$t_4[\text{dni_invitado}] = t_2[\text{dni_invitado}] = \{11222222\}$$

$$t_4[R-X-Y] = t_1[R-X-Y]$$

$$t_4[\text{nom_contr}, \text{serv_contr}] = t_1[\text{nom_contr}, \text{serv_contr}] = \{\text{Pedro Guti}, \text{empanadas}\}$$

Si bien las dependencias multivaluadas se hallan a partir de la semántica de los atributos, el ejemplo anterior con datos particulares, ha sido incluido para que se pueda comprender a que refiere la definición de dependencia multivaluada, que contempla la forma de las tuplas.



Teoría de diseño de BBDD relacionales

F10(#salon, fecha_fiesta, nom_contratante, servicio_contratado,
dni_invitado)

- En cada salón se realiza una sola fiesta por día, en un día puede haber varias fiestas en diferentes salones.
- Para cada fiesta puede figurar más de un contratante
- Cada invitado a una fiesta tiene asociado un número de mesa
- La cantidad de mesas del salón varia para cada fiesta
- Servicio contratado es una lista que describe los tipos de comida contratados para una fiesta (Esto quiere decir que para cada comida de la fiesta, aparece una tupla diferente en el esquema)
- Una persona puede ir a más de una fiesta en el mismo salón en diferentes fechas
- Una persona puede ir a mas de una fiesta el mismo día

En F10, dijimos que valía la DM: #salon, fecha_fiesta -->> DNI_invitado

¿Qué otras dependencias multivaluadas valen en F10?

Teoría de diseño de BBDD relacionales

F10(#salon, fecha_fiesta, nom_contratante, servicio_contratado,
dni_invitado)

- En cada salón se realiza una sola fiesta por día, en un día puede haber varias fiestas en diferentes salones.
- Para cada fiesta puede figurar más de un contratante
- Cada invitado a una fiesta tiene asociado un número de mesa
- La cantidad de mesas del salón varia para cada fiesta
- Servicio contratado es una lista que describe los tipos de comida contratados para una fiesta (Esto quiere decir que para cada comida de la fiesta, aparece una tupla diferente en el esquema)
- Una persona puede ir a más de una fiesta en el mismo salón en diferentes fechas
- Una persona puede ir a mas de una fiesta el mismo día

#salon, fecha_fiesta ->> servicio_contratado

#salon, fecha_fiesta ->> nom_contratante

Teoría de diseño de BBDD relacionales

Particionamiento de un esquema
por existencia de DMs no triviales



Teoría de diseño de BBDD relacionales

F10(#salon fecha_fiesta, nom_contratante, servicio_contratado,
dni_invitado)

Dependencias Multivaluadas válidas sobre F10:

DM1) #salon, fecha_fiesta \twoheadrightarrow servicio_contratado

DM2) #salon, fecha_fiesta \twoheadrightarrow nom_contratante

DM3) #salon, fecha_fiesta \twoheadrightarrow dni_invitado

El esquema F10 no está en 4NF porque existen dependencias multivaluadas 1-3 y no son triviales en F10. Entonces se particiona F10 utilizando una de las dependencias multivaluadas, por ejemplo DM1

Teoría de diseño de BBDD relacionales

F10(#salon fecha_fiesta, nom_contratante, servicio_contratado,
dni_invitado)

Dependencias Multivaluadas válidas sobre F10:

DM1) #salon, fecha_fiesta ->> servicio_contratado

DM2) #salon, fecha_fiesta ->> nom_contratante

DM3) #salon, fecha_fiesta ->> dni_invitado

F11(#salon, fecha_fiesta, servicio_contratado)

F12(#salon, fecha_fiesta, nom_contratante,
dni_invitado)

Teoría de diseño de BBDD relacionales

F11(#salon, fecha_fiesta, servicio_contratado)

F12(#salon, fecha_fiesta, nom_contratante, dni_invitado)

La partición F11 esta en 4NF ya que sólo vale la dependencia multivaluada 1 que es trivial en ella.

La particion F12 no esta en 4NF porque existen las dependencias multivaluadas:

DM2) #salon, fecha_fiesta \twoheadrightarrow nom_contratante

DM3) #salon, fecha_fiesta \twoheadrightarrow dni_invitado

y éstas no son triviales en F12.

Entonces divido la partición F12 utilizando la dependencia multivaluada 2 quedando lo siguiente:

F13(#salon, fecha_fiesta, nom_contratante)

F14(#salon, fecha_fiesta, dni_invitado)

Teoría de diseño de BBDD relacionales

F13(#salon, fecha_fiesta, nom_contratante)

F14(#salon, fecha_fiesta, dni_invitado)

La partición F13 está en 4NF ya que sólo vale la dependencia multivaluada 2 que es trivial en ella.

La partición F14 está en 4NF porque existe solamente la dependencia multivaluada DM3 que es trivial en ella.

Teoría de diseño de BBDD relacionales

En síntesis



Resumiendo el proceso de normalización hasta 4FN

1. Encontrar las dependencias funcionales
2. Encontrar las claves candidatas
3. Verificar si el esquema cumple con la definición de BCNF, si no se cumple descomponer la relación sin perder información ni dependencias funcionales
 - Si se pierden dependencias funcionales, llevar a 3NF
 - Esto hasta dejar las particiones en BCNF o 3FN
4. Expresar en este punto, cual es la clave primaria y que particiones quedaron en BCNF o 3FN (según corresponda)
5. Encontrar las dependencias multivaluadas sobre la última partición realizada (aquella que tiene la clave primaria del esquema) y verificar 4NF, si no se cumple dividir la relación
 - Esto se hace hasta dejar las particiones del esquema en 4FN
6. Expresar las particiones resultantes que quedaron en 4FN
 - Explicar porque las particiones descritas en el ítem 4 (excepto la analizada en el punto 5) quedaron en 4FN
7. Indicar que particiones en 4FN quedan en el esquema final (que no sean proyecciones de atributos claves de otras particiones en 4FN)

Ejercicio para analizar
individualmente para la próxima
clase



Teoría de diseño de BBDD relacionales – EJERCICIO–

RECITALES(idRecital, idGrupo, idIntegranteGrupo, marcaInstrumento, vocalista, idOrganizador, nombreGrupo, nombreOrganizador)

Donde

- ▶ en un recital se presentan diversos grupos y un grupo se presenta en diversos recitales
- ▶ cada grupo tiene diversos integrantes. Los integrantes del grupo pueden variar para diferentes recitales
- ▶ cada grupo tiene solamente un vocalista, el vocalista del grupo no varía para los diferentes recitales en los que el grupo se presenta
- ▶ de un grupo se conoce el nombre, pero puede suceder que dos grupos se llamen de igual manera
- ▶ de cada integrante de un grupo y para cada recital en los que el grupo se presenta, se sabe que marca de instrumento uso el integrante
- ▶ un integrante de un grupo puede pertenecer a más de un grupo
- ▶ un vocalista de un grupo puede ser vocalista de otros grupos
- ▶ cada recital tiene diversos organizadores y un organizador puede organizar diversos recitales
- ▶ el nombre del organizador se puede repetir para diferentes idOrganizador. El idOrganizador es único

Realizar el proceso de normalización para llevar RECITALES a 4FN

Algunos anuncios

▶ Miércoles 04 de Octubre

- Retomamos el ejercicio propuesto
- DBMS Relacional
- Para los interesados en la promoción
 - Segundo parcialito
 - Incluye la entrega de un ejercicio (a definir para el parcialito) que ya tiene que estar resuelto. En caso de referir a uno de los ejercicios de varios incisos, se solicitará un inciso en particular (por ejemplo, ejercicio 28 inciso m)

▶ Semana del 09 de octubre

- No hay actividades académicas
- **¡IMPORTANTE!**
 - CACIC
 - LACLO