

BASES DE DATOS
TEMA 5. DISEÑO DE BASES DE DATOS RELACIONALES
MEDIANTE NORMALIZACION

Contenidos generales

- * Redundancia y Anomalías en la actualización
- * Dependencias funcionales
- * Formas normales. Proceso de normalización

Motivación

BDR tienen un esquema formado por un conjunto de tablas
ER como herramienta de modelado
Transformación ER a tablas
Normalización: Modelado específico para BDR
Basado en Dependencias funcionales
Medida formal para la calidad del diseño de BDR

5.1. Introducción

Normalización: Método de diseño de BDR

Introducida por Codd en los 70

Basado en fundamentos teóricos

Parte de todo el conjunto de atributos, y mediante su análisis y proceso obtiene un esquema relacional

Proceso descendente

Idea: Cada hecho almacenado una sola vez

5.2. Información redundante y Anomalías en la actualiz. (1)

Objetivos: Reducir el espacio de almacenamiento
Evitar anomalías

Ejemplo: Empleados y Departamentos

EMPLEADO_DPTO (NombreEmp, NumSS, FechaEntrada, Direccion, NumDep, NombreDep, NumSSJefe)

Problema 1: Información de departamentos repetida

NombreEmp	NumSS	FechaEntrada	Direccion	NumDep	NombreDep	NumSSJefe
Javier Ros	1	10-10-1990	Azul 6	010	Ventas	5
Isabel Gil	2	16-12-1991	Gris 5	010	Ventas	5
Inma Sol	5	1-2-1990	Jaspe 3	010	Ventas	-
Jose Leal	7	6-7-1998	Pais 45	011	Compras	7

Problema 2: Semántica compleja

5.2. Información redundante y Anomalías en la actualiz. (2)

Esquema alternativo

EMPLEADO (NombreEmp, NumSS, FechaEntrada, Direccion, NumDep)
DPTO (NumDep, NombreDep, NumSSJefe)

Ventajas:

- Datos de departamento almacenados una sola vez
- Semántica sencilla
- Evita la redundancia
- Evita anomalías en la actualización
 - Anomalías de inserción
 - Anomalías de eliminación
 - Anomalías de modificación

5.2. Información redundante y Anomalías en la actualiz. (3)

NombreEmp	NumSS	FechaEntrada	Direccion	NumDep	NombreDep	NumSSJefe
Javier Ros	1	10-10-1990	Azul 6	010	Ventas	5
Isabel Gil	2	16-12-1991	Gris 5	010	Ventas	5
Inma Sol	5	1-2-1990	Jaspe 3	010	Ventas	-
Jose Leal	7	6-7-1998	Pais 45	011	Compras	7

EMPLEADOS

NombreEmp	NumSS	FechaEntrada	Direccion	NumDep	NumSSJefe
Javier Ros	1	10-10-1990	Azul 6	010	5
Isabel Gil	2	16-12-1991	Gris 5	010	5
Inma Sol	5	1-2-1990	Jaspe 3	010	-
Jose Leal	7	6-7-1998	Pais 45	011	

DPTO

NumDep	NombreDep
010	Ventas
011	Compras

5.2. Información redundante y Anomalías en la actualiz. (4)

5.2.1. Anomalías de inserción

Dos tipos

- 1) Insertar un empleado -> Insertar un departamento
 - Valores nulos si no hay departamento
 - Valores de departamento concordantes (Consistencia)
- 2) Insertar un departamento -> Insertar un empleado
 - Personal no asignado aún a departamentos ??

5.2. Información redundante y Anomalías en la actualiz. (5)

5.2.2. Anomalías de eliminación

Eliminar una fila del último empleado de un departamento
elimina los datos de dicho departamento (p.e. Eliminar
José Leal)

5.2.3. Anomalías de modificación

Modificar datos en una fila puede implicar modificarlo en
varias filas para evitar inconsistencia
Valores nulos deberían tener un significado (NA, UNK, ...)

5.3. Dependencias funcionales (1)

DF: Restricción entre dos conjuntos de atributos

$$R = \{A_1, A_2 \dots A_n\}$$
$$X, Y \subseteq R$$

Y depende funcionalmente de X ($X \rightarrow Y$)

$$\forall t_1, t_2 \in \text{ext}(R): t_1[X] = t_2[X], \Rightarrow t_1[Y] = t_2[Y]$$

Ejemplo

EMPLEADO (NombreEmp, NumSS, FechaEntrada, Direccion, NumDep)

{NumSS} \rightarrow {NombreEmp, FechaEntrada, Direccion, NumDep}

{NombreEmp} $\not\rightarrow$ {FechaEntrada, Direccion}

Claves candidatas determinan de forma única el resto de atributos de una relación

DF dependen del esquema y no de los datos

5.3. Dependencias funcionales (2)

Dependencia funcional transitiva

$$X \rightarrow Y \text{ si } \exists Z / X \rightarrow Z \wedge Z \rightarrow Y$$

Ejemplo

(NombreEmp, NumSS, FechaEntrada, Direccion, NumDep, NombreDep, NumSSJefe)

{NumSS} \rightarrow {NombreDep} (Transitiva)

{NumSS} \rightarrow {NumDep}

{NumDep} \rightarrow {NombreDep}

5.4. Normalización basada en claves primarias (1)

Normalización: Proceso de diseño para BDR
Basada en dependencias funcionales
Proceso de descomposición descendente
Codd: 1FN, 2FN, 3FN

Produce descomposición sin pérdida
Conservación de dependencias

Superclave

Clave candidata

Clave primaria

Atributos primos: Miembros de claves candidatas

Atributos no primos: No pertenecen a claves candidatas

5.4. Normalización basada en claves primarias (2)

5.4.1. Primera forma normal (1FN)

Dominios de atributos con valores indivisibles (atómicos)
No se permiten atributos multivaluados ni relaciones
anidadas

Def: Relación en 1FN si sus atributos no tienen dominios
que sean conjuntos

`R = (Num_Pedido, Fecha, Num_Proveedor, Nombre_Proveedor,
Dirección_Proveedor, Num_Producto, Descripción, Precio,
Cantidad, Precio_Total_Producto, Precio_Total_Pedido)`

Multivaluados:

`Num_Producto, Descripción, Precio, Cantidad,
Precio_Total_Producto`

5.4. Normalización basada en claves primarias (3)

5.4.1. Primera forma normal (1FN)

Solución:

- a) Descomponer (Considera que es una tabla de Pedidos)
- b) Ampliar la clave (Considera que es una tabla de Productos en Pedidos)

Solución a)

```
R = (Num Pedido, Fecha, Num_Proveedor, Nombre_Proveedor,
Dirección_Proveedor, Num_Producto, Descripción, Precio,
Cantidad, Precio Total_Producto, Precio_Total_Pedido)
```

```
Pedidos = (Num Pedido, Fecha,
Num_Proveedor, Nombre_Proveedor,
Dirección_Proveedor,
Precio_Total_Pedido)
```

```
ProductosPedido =
(Num Pedido, Num Producto,
Descripción, Precio,
Cantidad, Precio
Total_Producto)
```

5.4. Normalización basada en claves primarias (4)

5.4.1. Primera forma normal (1FN)

Solución b)

```
R = (Num Pedido, Fecha, Num_Proveedor, Nombre_Proveedor,
Dirección_Proveedor, Num_Producto, Descripción, Precio,
Cantidad, Precio Total_Producto, Precio_Total_Pedido)
```

```
ProductosPedido = (Num Pedido, Fecha, Num_Proveedor,
Nombre_Proveedor, Dirección_Proveedor, Num Producto,
Descripción, Precio, Cantidad, Precio Total_Producto,
Precio_Total_Pedido)
```

5.4. Normalización basada en claves primarias (5)

5.4.2. Segunda forma normal (2FN)

Elimina dependencias de partes de la clave primaria

Def: Relación en 2FN si está en 1FN y las dependencias funcionales son totales

En la solución a) ProductosPedido no está en 2FN

Dependencias parciales

```
{Num-Producto} -> {Descripción, Precio}
```

En la solución b) ProductosPedido no está en 2FN

Dependencias parciales

```
{Num_Pedido} -> {Fecha, Num_Proveedor, Nombre_Proveedor,  
Dirección_Proveedor, Precio_Total_Pedido}  
{Num-Producto} -> {Descripción, Precio}
```

5.4. Normalización basada en claves primarias (6)

5.4.2. Segunda forma normal (2FN)

Solución:

Crear nuevas relaciones con las dependencias parciales

Para la solución a)

```
ProductosPedido = (Num Pedido,  
Num Producto, Descripción,  
Precio, Cantidad,  
Precio_Total_Producto)
```

```
ProductosPedido =  
(Num Pedido, Num Producto,  
Cantidad,  
Precio_Total_Producto)
```

```
Productos = (Num Producto,  
Descripción, Precio)
```

5.4. Normalización basada en claves primarias (7)

5.4.2. Segunda forma normal (2FN)

Para la solución b)

```
ProductosPedido = (Num Pedido, Fecha, Num_Proveedor,  
Nombre_Proveedor, Dirección_Proveedor, Num Producto,  
Descripción, Precio, Cantidad, Precio Total_Producto,  
Precio_Total_Pedido)
```

```
ProductosPedido =  
(Num Pedido, Num Producto,  
Cantidad, Precio  
Total_Producto)
```

```
Productos = (Num Producto,  
Descripción, Precio)
```

```
Pedidos = (Num Pedido,  
Fecha, Num_Proveedor,  
Nombre_Proveedor,  
Dirección_Proveedor,  
Precio_Total_Pedido)
```

5.4. Normalización basada en claves primarias (8)

5.4.3. Tercera forma normal (3FN)

Elimina dependencias transitivas

Solución: Crear nueva relación con atributos con la dependencia transitiva

En a) y b) Pedidos no está en 3FN

Dependencia transitiva:

```
{Num_Pedido} -> {Num_Proveedor}
```

```
{Num_Proveedor} -> {Nombre_Proveedor, Dirección_Proveedor}
```

5.4. Normalización basada en claves primarias (9)

5.4.3. Tercera forma normal (3FN)

Para las soluciones a) y b)

```
Pedidos = (Num Pedido, Fecha, Num_Proveedor,  
Nombre_Proveedor, Dirección_Proveedor, Precio_Total_Pedido)
```

```
Pedidos = (Num Pedido, Fecha,  
Num_Proveedor,  
Precio_Total_Pedido)
```

```
Proveedores = (Num Proveedor,  
Nombre_Proveedor,  
Dirección_Proveedor)
```