



Dpto. Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur

## ELEMENTOS DE BASES DE DATOS

Segundo Cuatrimestre 2013

### Clase 10: Teoría Modelo Relacional – Formas Normales - Normalización

Mg. María Mercedes Vitturini  
[mvitturi@uns.edu.ar]



## Forma Normal

La **forma normal** es una condición sobre un esquema de base de datos en función de los esquemas de relación, sus dependencias funcionales y sus llaves candidatas.

- Las formas normales que vamos a estudiar son:
  - Primer Forma Normal (1FN),
  - Segunda Forma Normal (2FN),
  - Tercer Forma Normal (3FN), y
  - Forma Normal de Boyce Codd (FNBC).

EBD2012\_10 - Mg. Mercedes Vitturini

## Normalización

La **normalización** es un proceso que sucesivamente reduce esquemas de relación con anomalías para producir nuevos esquemas mejor estructurados.

- Objetivos de la normalización
  - Minimizar la redundancia de datos.
  - Simplificar la gestión de datos (ABM).
  - Proveer un diseño que favorece el crecimiento de la información.
- En general, una base de datos normalizada no favorece:
  - La simplicidad de las consultas.
  - El tiempo de acceso a la información.

EBD2012\_8 - Mg. Mercedes Vitturini

## Primera Forma Normal (1FN)

**Definición:** Un esquema de relación R está en **primera forma normal (1FN)** si los valores en  $\text{dom}(A)$  son atómicos para todo atributo A en R.

- Un **dominio es atómico** si sus elementos se pueden considerar como unidades indivisibles.
  - Exige que los valores para los dominios **no** sean valores compuestos (conjuntos, listas, etc.).
- Características de la 1FN:
  - Permite representación directa en tablas.
  - Lenguajes de consulta más simples.
  - Es posible definir restricciones o dependencias funcionales entre sus atributos.

EBD2012\_10 - Mg. Mercedes Vitturini

## Ejemplos – No cumplen 1FN

- Personas** = (DNI, Apellido, Nombres, Domicilio,  $_o\{\text{Hobby}\}_n$ )
  - El grupo que se repite de hobby **no respeta la 1FN.**
- Artículos** = (código, nombre, descripción, nombDpto)
  - Con *dom* (código) valores de la forma dd-cccc, donde dd representa el departamento que lo comercializa y cccc el número interno de artículo, **no respeta la 1FN.**

EBD2012\_10 - Mg. Mercedes Vitturini

## Definiciones previas

**Definición** – Dados un esquema de relación R, un atributo A en R y un conjunto de df's F, un atributo **A es primo en R con respecto a F** si **A es parte de alguna llave candidata de R**. De lo contrario, se dice que **A es no primo.**

**Definición** – Dados un conjunto de df's F y una df  $X \rightarrow Y$ , se dice que **Y es parcialmente dependiente de X** si en F,  $X \rightarrow Y$  no está reducido a izquierda, esto es, **Y depende de un subconjunto de los atributos de X**. Si  $X \rightarrow Y$  está reducido a izquierda entonces **Y es totalmente dependiente de X.**

EBD2012\_10 - Mg. Mercedes Vitturini

## Segunda Forma Normal (2FN)

**Definición #1:** Un esquema de relación R está en **segunda forma normal (2FN)** con respecto a un conjunto de df's F si está en 1FN y cada atributo no primo es totalmente dependiente de cada llave candidata en R.

**Definición #2:** Un esquema de relación R está en **segunda forma normal (2FN)** con respecto a un conjunto de df's F si para cada df  $X \rightarrow A$  ( $A \notin X$ ) en F se verifica que:

- X no es un subconjunto propio de una llave, o bien:
- A es primo.

*2FN admite df entre no primos*

EBD2012\_10 - Mg. Mercedes Vitturini

## 2FN – Ejemplo

**Funciones (funNro, fecha, espectador, asientoNro)**

(en forma reducida  $R(N,F,E,A)$ )

- Sobre R se definen el siguiente conjunto F de df's:

$$F = \{ NE \rightarrow A, N \rightarrow F \}$$

- **Llave:** NE, con
  - N y E son atributos **primos**,
  - A y F son **no primos**.
- **Funciones no respeta 2FN.** El atributo no primo F es parcialmente dependiente de la llave NE.

EBD2012\_10 - Mg. Mercedes Vitturini

## Problemas del esquema

**Funciones (funNro, fecha, espectador, asientoNro)**

- **Funciones** tiene los siguientes problemas de diseño:
  - Se repite la información número de función con su fecha en tantas tuplas como espectadores tenga asociados la función.
  - Para agregar una función se necesita tener definido al menos un espectador.
  - Si el último espectador que asiste a una función se borra, se pierde información sobre la fecha de emisión de la función.
  - En caso de cambiar la fecha de una función se debería cambiar en todos los espectadores asociados a la misma.

EBD2012\_10 - Mg. Mercedes Vitturini

## Una mejor organización

- Para R (NFEA) y  $F = \{ NE \rightarrow A, N \rightarrow F \}$ .

- La descomposición:

$$\rho = (R_1, R_2) \text{ donde } R_1(\text{NEA}) \text{ y } R_2(\text{NF})$$

- $R_1 = (\text{NEA})$ ,  $F_1 = \{ NE \rightarrow A \}$  llave NE, satisface la 2FN.
- $R_2 = (\text{NF})$ ,  $F_2 = \{ N \rightarrow F \}$  llave N, satisface la 2FN.

- Dado que cada subesquema satisface la 2FN, luego la descomposición  $\rho$  satisface la 2FN.
- ¿Qué puede decir de esta descomposición con relación a las propiedades JSP y PD?

EBD2012\_10 - Mg. Mercedes Vitturini

## Ejercicios

- Dados los siguientes esquemas y sus df's, determinar si satisfacen 2FN:

a. **Libros**(ISBN, Título, Nombre\_Autor)  $\equiv L(I,T,A)$

$$F = \{ I \rightarrow T \}$$

b. **Cientes** = (CliNro, CliNombre, CliDomicilio, CódPostal, NomLocalidad)  $\equiv C(C,N,D,P,L)$

$$F = \{ C \rightarrow NDP, P \rightarrow L \}$$

EBD2012\_10 - Mg. Mercedes Vitturini

## Tercera Forma Normal (3FN)

**Definición #1:** Un esquema de relación R está en **tercera forma normal (3FN)** con respecto a un conjunto de df's F si está en 1FN y cada atributo no primo NO es transitivamente dependiente de una superllave de R.

**Definición #2:** Un esquema de relación R está en **tercera forma normal (3FN)** si para cada dependencia funcional  $X \rightarrow A$  ( $A \notin X$ ) en F se verifica que:

- X es llave, o bien,
- A es primo.

*3FN admite df entre primos*

EBD2012\_10 - Mg. Mercedes Vitturini

### 3FN – Ejemplo

Dado:

**Vuelos**(VueloNro, Fecha, PilotoNro, NombPiloto)  
 $F = \{VF \rightarrow P, P \rightarrow N\}$

- Llave **VF** con
  - **V, F** atributos **primos** y
  - **P, N** son **no primos**.
  - **VUELOS** **respet**a 2FN, **pero no respeta** en 3FN.
- La descomposición de R:
  - $\rho = (R_1, R_2)$  con  $R_1(VFP)$   $R_2(PN)$
- y las df's  $\{VF \rightarrow P\}$  en  $R_1$  y  $\{P \rightarrow N\}$  en  $R_2$  satisface 3FN.
- ¿Es JSP y PD?

EBD2012\_10 - Mg. Mercedes Vitturini

### Problemas de Diseño

**Vuelos (VueloNro, Fecha, PilotoNro, NombPiloto)**

- Vuelos tiene los siguientes problemas de diseño:
  - Sólo se posible mantener información de un piloto si está asociado a un vuelo.
  - Se mantiene la información de un piloto y su nombre tantas veces como vuelos tenga asignados.
  - Si fuese necesario modificar el nombre del piloto, debería corregirse en todos los vuelos a los que fue asignado.

EBD2012\_10 - Mg. Mercedes Vitturini

### Forma Normal de Boyce-Codd (FNBC)

**Definición:** Un esquema de relación R está en **forma normal de Boyce-Codd (FNBC)** con respecto a un conjunto de df's  $F$  si para cada dependencia funcional  $X \rightarrow A$  ( $A \notin X$ , esto es **no trivial**) se verifica que X es llave.

- Ejemplo:
  - **Clientes = (ClienteNro, Nombre, Domicilio, FechaNac)**
  - $\equiv C (C, N, D, F)$
  - $F = \{C \rightarrow N, D, F\}$

*FNBC la única df no trivial es de la llave*

EBD2012\_10 - Mg. Mercedes Vitturini

### Descomposiciones normalizadas

- Los almacenamientos (o esquemas de relación) que no respetan la **3FN no responden a un buen diseño relacional de datos**. Tienen problemas diseño que se traducen en:
  - Redundancia de información.
  - Inconsistencias.
  - Anomalías de inserción y de borrado.
- La solución a un problema de diseño de datos es encontrar una **descomposición** mejor.

EBD2012\_10 - Mg. Mercedes Vitturini

### Descomposiciones normalizadas

- Supongamos ahora la siguiente definición para un esquema de relación **Dicta**, que mantiene información de los docentes y las materias que dictan cada año:

**Dicta** (prof-legajo, prof-apellido, prof-nombre, prof-domicilio, {mat-código, dicta-año, dicta-cuat, mat-nombre, dpto-código, dpto-nombre}<sub>n</sub>)

$F = \{\text{prof-legajo} \rightarrow \text{prof-apellido, prof-nombre, prof-domicilio};$   
 $\text{mat-código} \rightarrow \text{mat-nombre, dpto-código};$   
 $\text{dpto-código} \rightarrow \text{dpto-nombre}\}$

EBD2012\_10 - Mg. Mercedes Vitturini

### Primer Forma Normal

- El esquema **Dicta NO RESPETA LA 1FN:**
  - Existen atributos **no atómicos**.
- En este caso por cada docente se mantiene un grupo no atómico con las materias que el profesor dicta y/o ha dictado en distintos cuatrimestres.
- Para contar con un modelo de datos que respete la **1FN**, la solución es descomponer.

EBD2012\_10 - Mg. Mercedes Vitturini

## Descomposición en 1FN

- Si un esquema no respeta la 1FN se debe descomponer en dos esquemas:
  1. Uno con todos los datos que no pertenecen al grupo repetitivo.
  2. Uno con el grupo que se repite al que se le agrega la clave del esquema del inciso anterior.

EBD2012\_10 - Mg.  
Mercedes Vitturini

## Descomposición en 1FN

- **Dicta<sub>1</sub>** (prof-legajo, prof-apellido, prof-nombre, prof-domicilio).
- **Dicta<sub>2</sub>** (prof-legajo, mat-código, dicta-año, dicta-cuat, mat-nombre, dpto-código, dpto-nombre)

EBD2012\_10 - Mg.  
Mercedes Vitturini

## Esquema Dicta<sub>1</sub>

- **Dicta<sub>1</sub>** (prof legajo, prof-apellido, prof-nombre, prof-domicilio)
- **Dependencias Funcionales:**
  - $F_1 = \{\text{prof-legajo} \rightarrow \text{prof-apellido, prof-nombre, prof-domicilio}\}$
- **Clave:** prof-legajo.
- **Atributos primos:** prof-legajo.
- **Atributos no primos:** prof-apellido, prof-nombre, prof-domicilio.

EBD2012\_10 - Mg.  
Mercedes Vitturini

## Esquema Dicta<sub>2</sub>

- **Dicta<sub>2</sub>** (prof-legajo, mat-código, dicta-año, dicta-cuat, mat-nombre, dpto-código, dpto-nombre)
- **Dependencias Funcionales:**
  - $F_2 = \{\text{mat-código} \rightarrow \text{mat-nombre, dpto-código}; \text{dpto-código} \rightarrow \text{dpto-nombre}\}$
- **Clave:** prof-legajo + mat-código + dicta-cuat + dicta-año.
- **Atributos primos:** prof-legajo, mat-código, dicta-cuat, dicta-año
- **Atributos no primos:** mat-nombre, dpto-código, dpto-nombre

EBD2012\_10 - Mg.  
Mercedes Vitturini

## Segunda Forma Normal

- **Dicta<sub>1</sub>** respeta la **2FN** ya que la clave esta formada por un único atributo.
- **Dicta<sub>1</sub>** (prof-legajo, prof-apellido, prof-nombre, prof-domicilio).

EBD2012\_10 - Mg. Mercedes Vitturini

## Segunda Forma Normal

**Dicta<sub>2</sub>** NO RESPETA la **2FN**.

**Dicta<sub>2</sub>** (prof-legajo, mat-código, dicta-cuat, dicta-año, mat-nombre, dpto-código, dpto-nombre)

- **Clave:** prof-legajo, mat-código, dicta-cuat, dicta-año.
- Existe una dependencia funcional:
  - **mat-código** → **mat-nombre, dpto-código, dpto-nombre**
  - Los **atributos no primos:** mat-nombre, dpto-código y dpto-nombre *dependen parcialmente de la llave.*

EBD2012\_10 - Mg.  
Mercedes Vitturini

## Descomposición en 2FN

- La solución para un almacenamiento que no respeta la **2FN** es *descomponer*:
  1. En un almacenamiento con la llave primaria y todos aquellos atributos que dependen totalmente de la llave primaria (si los hubiera).
  2. Otro almacenamiento con los atributos que dependen de parte de la llave, junto con la parte de la llave que ocasiona la dependencia.

EBD2012\_10 - Mg. Mercedes Vitturini

## Descomposición en 2FN

- $DictA_2 = (\text{prof-legajo, mat-código, dicta-cuat, dicta-año, mat-nombre, dpto-código, dpto-nombre})$ 
  - $DictA_{21} (\text{prof-legajo, mat-código, dicta-cuat, dicta-año})$
  - $DictA_{22} (\text{mat-código, mat-nombre, dpto-código, dpto-nombre})$ .

EBD2012\_10 - Mg. Mercedes Vitturini

## Esquema Dicta<sub>21</sub>

$DictA_{21}(\text{prof-legajo, mat-código, dicta-cuat, dicta-año})$

- **Dependencias Funcionales:** no posee.
- **Clave:** prof-legajo + mat-código + dicta-cuat + dicta-año.
- **Atributos primos:** todos.
- **Atributos no primos:** no posee.

EBD2012\_10 - Mg. Mercedes Vitturini

## Esquema Dicta<sub>22</sub>

$DictA_{22}(\text{mat-código, mat-nombre, dpto-código, dpto-nombre})$

- **Dependencias Funcionales:**
  - $F_{22} = \{\text{mat-código} \rightarrow \text{mat-nombre, dpto-código}; \text{dpto-código} \rightarrow \text{dpto-nombre}\}$
- **Clave:** mat-código.
- **Atributos primos:** mat-código.
- **Atributos no primos:** mat-nombre, dpto-código, dpto-nombre.

EBD2012\_10 - Mg. Mercedes Vitturini

## Tercer Forma Normal

- El almacenamiento  $DictA_{21}$  respeta la **3FN** ya que no posee ningún atributo que no forme parte de la llave, por lo tanto no existen dependencias entre atributos no primos.

$DictA_{21}(\text{prof legajo, mat código, dicta año, dicta cuat})$

EBD2012\_10 - Mg. Mercedes Vitturini

## Tercer Forma Normal

$DictA_{22}$  NO RESPETA 3FN.

- Existe una dependencia funcional entre los atributos no primos *dpto-código* y *dpto-nombre*.
- Ninguno de ellos forma parte de la llave *mat-código*.
- La solución está en descomponer al almacenamiento  $DictA_{22}$

EBD2012\_10 - Mg. Mercedes Vitturini

## Descomposición 3FN

- Si *un esquema no respeta la 3FN, corresponde descomponerlo* en dos esquemas.
  1. Un esquema con la clave primaria y los atributos que dependen totalmente de la clave primaria (sin incluir el que depende transitivamente).
  2. Otro esquema con los atributos participan de la dependencia que viola la 3FN.
- En el ejemplo:

EBD2012\_10 - Mg. Mercedes Vitturini

## Descomposición en 3FN

- **Dicta22** (mat-código, mat-nombre, dpto-código, dpto-nombre)
- **Dicta<sub>221</sub>** (mat-código, mat-nombre, dpto-código)
- **Dicta<sub>222</sub>** (dpto-código, dpto-nombre)

EBD2012\_10 - Mg. Mercedes Vitturini

## Esquema Dicta<sub>221</sub>

**Dicta<sub>221</sub>** (mat-código, mat-nombre, dpto-código)

- **Dependencias Funcionales:**
  - $F_{221} = \{ \text{mat-código} \rightarrow \text{mat-nombre, dpto-código} \}$
- **Clave:** mat-código
- **Atributos primos:** mat-código
- **Atributos no primos:** mat-nombre, dpto-código.

EBD2012\_10 - Mg. Mercedes Vitturini

## Esquema Dicta<sub>222</sub>

**Dicta<sub>222</sub>** (dpto-código, dpto-nombre)

- **Dependencias Funcionales:**
  - $F_{222} = \{ \text{dpto-código} \rightarrow \text{dpto-nombre} \}$
- **Clave:** dpto-código.
- **Atributos primos:** dpto-código
- **Atributos no primos:** dpto-nombre.

EBD2012\_10 - Mg. Mercedes Vitturini

## Descomposición en 3FN

$\rho$  (Dicta<sub>1</sub>, Dicta<sub>21</sub>, Dicta<sub>221</sub>, Dicta<sub>222</sub>) donde

- **Dicta<sub>1</sub>** (prof-legajo, prof-apellido, prof-nombre, prof-domicilio)  
 $F_1 = \{ \text{prof-legajo} \rightarrow \text{prof-apellido, prof-nombre, prof-domicilio} \}$
- **Dicta<sub>21</sub>** (prof-legajo, mat-código, dicta-año, dicta-cuat)  
 $F_{21} = \{ \emptyset \}$
- **Dicta<sub>221</sub>** (mat-código, mat-nombre, dpto-código)  
 $F_{221} = \{ \text{mat-código} \rightarrow \text{mat-nombre, dpto-código} \}$
- **Dicta<sub>222</sub>** (dpto-código, dpto-nombre)  
 $F_{222} = \{ \text{dpto-código} \rightarrow \text{dpto-nombre} \}$

EBD2012\_10 - Mg. Mercedes Vitturini

## Descomposición en 3FN con nombres significativos

- **PROFESOR** = (prof-legajo, prof-apellido, prof-nombre, prof-domicilio)
  - **Dicta** = (prof-legajo, mat-código, dicta-año, dicta-cuat)
  - **MATERIA** = (mat-código, mat-nombre, dpto-código)
  - **DEPARTAMENTO** = (dpto-código, dpto-nombre)
- $\rho$  (PROFESOR, Dicta, MATERIA, DEPARTAMENTO)

EBD2012\_10 - Mg. Mercedes Vitturini

## FNBC

- En el ejemplo:  
**p (PROFESOR, DICTA, MATERIA, DEPARTAMENTO)**  
cada subesquema  $R_i$  tiene como única dependencia funcional no trivial  $X \rightarrow Y$ , con X llave para  $R_i$ , por lo tanto **p** está en FNBC.
- Dado un esquema R, si R no está en una buena forma normal (3FN o FNBC):
  - *Siempre* es posible encontrar una descomposición **3FN, pd, jsp**.
  - *Siempre* es posible encontrar una descomposición **FNBC, jsp**.

EBD2012\_10 - Mg. Mercedes Vitturini

## ¿Cómo se diseña el modelo de datos?

- Diseñar el modelado conceptual
  - Modelo Entidad Relación (MER)
  - El Diagrama Entidad Relación (DER)

El traslado de un DER al MR se hace siguiendo reglas

- Diseñar el modelo lógico
  - Modelo Relacional (MR)
- Revisar el diseño del MR
  - Normalizar (descomponer)

Evolución en el diseño de modelos

Elementos básicos: conjuntos de entidades y conjuntos de relaciones.

Elemento básico: esquemas de relaciones. Los esquemas con sus dependencias funcionales se analizan para ver que respeten al menos 3FN, sino se descomponen.

EBD2012\_10 - Mg. Mercedes Vitturini

## Temas de la clase de hoy

- Conceptos de Normalización de almacenamientos
  - Dependencias funcionales.
  - Claves de almacenamiento.
  - Atributos primos y no primos.
- Formas normales (1FN, 2FN, 3FN y FNBC)
- **Bibliografía:**
  - *Database System Concepts* – Abraham Silberschatz
  - Capítulo 7

EBD2012\_10 - Mg. Mercedes Vitturini