

# Normalización

- Licenciatura e Ingeniería en Sistemas
- 2do. año

# Normalización

- Proceso en el que se transforma un esquema de relación en otro con buenas propiedades:
  - Baja redundancia.
  - Nulos controlados.
  - Sin generación de tuplas erróneas.
- Se puede basar en dependencias funcionales o en otros tipos de dependencias.

# Normalización = Descomposición ¿de cualquier forma?

- La idea básica de normalización es el reemplazo de un esquema de relación por una descomposición en una forma normal más alta.
- ¿Sirve cualquier descomposición?

# Ejemplo

- Univ(CodMat, CiEst, IdGrupo, HrsSem, Cred)
- SC={CodMat→HrsSem;  
CiEst→IdGrupo; HrsSem→Cred}

# Ejemplo

- $SC = \{ \text{CodMat} \rightarrow \text{HrsSem}; \text{CiEst} \rightarrow \text{IdGrupo}; \text{HrsSem} \rightarrow \text{Cred} \}$

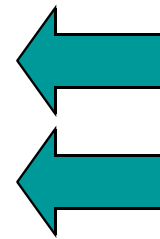
CodMat	CiEst	IdGrp	HrsS	Cred
$cm_1$	$ce_1$	$gr_1$	$h_1$	$cr_1$
$cm_2$	$ce_2$	$gr_2$	$h_2$	$cr_2$

CodMat	HrsS	CiEst	IdGrp	HrsS	Cred
$cm_1$	$h_1$	$ce_1$	$gr_1$	$h_1$	$cr_1$
$cm_2$	$h_2$	$ce_2$	$gr_2$	$h_2$	$cr_2$

# Ejemplo

- Observar que:
  - En cada tabla de la descomposición se cumple una dependencia.
  - El Join genera tuplas erróneas !!
  - Es Join Con Pérdida

CodMat	CiEst	IdGrp	HrsS	Cred
cm <sub>1</sub>	ce <sub>1</sub>	gr <sub>1</sub>	h <sub>1</sub>	cr <sub>1</sub>
cm <sub>1</sub>	ce <sub>2</sub>	gr <sub>2</sub>	h <sub>1</sub>	cr <sub>1</sub>
cm <sub>2</sub>	ce <sub>1</sub>	gr <sub>1</sub>	h <sub>2</sub>	cr <sub>2</sub>
cm <sub>2</sub>	ce <sub>2</sub>	gr <sub>2</sub>	h <sub>2</sub>	cr <sub>2</sub>



# Otro Ejemplo

- OK. Entonces partimos de esta forma:

CodMat	HrsS	IdGrp
$cm_1$	$h_1$	$gr_1$
$cm_2$	$h_2$	$gr_2$

CiEst	HrsS	Cred
$ce_1$	$h_1$	$cr_1$
$ce_2$	$h_2$	$cr_2$
$ce_1$	$h_2$	$cr_2$

# Conclusiones

- No alcanza sólo con las dependencias.
- Es necesario definir claramente los criterios con que se van a construir descomposiciones, teniendo en cuenta:
  - La no generación de tuplas erróneas.
  - La preservación o no de dependencias.



# Calidad de las Descomposiciones

Las descomposiciones ideales cumplen las siguientes dos condiciones:

- No se pierden atributos
- Tienen JSP.
- Tienen Preservación de Dependencias.

Nunca se pueden aceptar esquemas que tengan join con pérdida (no tengan JSP).

# Formas Normales

- Una forma normal, es un conjunto de condiciones que debe cumplir un esquema relacional para que se considere que es “bueno”.
- Se definen en función de las dependencias que pueden ser funcionales o de otro tipo.
- Se presentarán algoritmos basados en estrategias de descomposición que garantizan que ese esquema cumple determinada forma normal.

# Formas Normales

- Todas las formas normales se definen para un esquema de relación (tabla) y un determinado conjunto de dependencias.
- Un esquema relacional (BD) cumple con una determinada forma normal si la cumplen todos los esquemas de relación que contiene.

# Primera Forma Normal

- Un esquema de relación está en primera forma normal (1NF) si todos los dominios de los atributos son atómicos.
  - Si no está en 1NF entonces no está en el Modelo Relacional.

# Definiciones

- Atributo Primo: Un atributo A en ATR es primo si es miembro de alguna clave de la relación. De lo contrario es No Primo.
- Ejemplos:
  - $R(A,B,C,D,E)$ 
    - $AB \rightarrow CDE$
  - Por la dependencia, AB es clave, por lo tanto A y B son primos y C,D y E no lo son.

# Hacia la Segunda Forma Normal

- Dependencia Parcial
  - Se dice que una df  $X \rightarrow A$  es parcial si hay algún subconjunto  $Y$  de  $X$  tal que  $Y \rightarrow A$
  - Ejemplo:
    - $SC = \{ABC \rightarrow D, B \rightarrow D\}$ 
      - Entonces  $ABC \rightarrow D$  es parcial (existe una df reducida respecto de ella).

# Segunda Forma Normal

- Un esquema de relación  $R$  está en segunda forma normal (2NF) con respecto al conjunto de dependencias funcionales  $SC$  si no hay ningún atributo NO primo de  $R$  que dependa parcialmente de una clave.

## Ej: Segunda Forma Normal

- Emp-Proy(CI, NumP, Horas, NomE, NomP, LugarP)
- $SC = \{CI, NUMP \rightarrow Horas; CI \rightarrow NomE; NumP \rightarrow NomP, LugarP\}$ 
  - Busco la clave (CI, NUMP)
  - Emp-Proy no está en 2NF porque:
    - NomE no es primo
      - $CI \rightarrow NomE$  (parte de clave determina un atributo no primo)
- Emp-Proy está en 1NF solamente.



# Hacia la Tercera Forma Normal

- Dependencia Transitiva
  - Una df  $X \rightarrow Y$  sobre un esquema de relación  $R$  es transitiva si:
    - Existe  $Z \subseteq R$  tal que  $Z$  no está en ninguna clave de  $R$
    - Se cumple que  $X \rightarrow Z$  y  $Z \rightarrow Y$ .

# Tercera Forma Normal

- Un esquema de relación  $R$  está en tercera forma normal (3NF) con respecto a un conjunto de dependencias  $F$  si no hay atributos no primos que dependan transitivamente de una clave.
- $R$  está en 3NF con respecto a  $F$  si toda dependencia  $X \rightarrow A$  de  $F^+$  cumple que:
  - O bien  $X$  es superclave de  $R$ .
  - O bien  $A$  es primo.

# Ejemplo de Tercera Forma Normal

- Emp-Depto(NomE, CI, FechaN, Dir, NumD, NombreD, Cljefe)
- $SC = \{CI \rightarrow NomE, FechaN, Dir, NumD; NumD \rightarrow NomD, Cljefe\}$ 
  - Busco la clave (CI)
  - Emp-Depto no está en 3NF porque:
    - NumD es tal que  $CI \rightarrow NumD$  y  $NumD \rightarrow NomD$  y NomD no es primo.
  - Otra forma:
    - Dada  $NumD \rightarrow NomD$ , se cumple que NomD no es primo y NumD no es superclave (verificar).

# BCNF

- BCNF (Boyce Code NF):

Un esquema de relación RS con un conjunto SC que contiene dfs cumple con BCNF si:

- Para  $c/X \rightarrow Y$  (dentro de SC) tal que Y no pertenece a X, X es una superclave de RS.
- Ejemplo:
  - Dicta(est,curso,prof)
  - $SC = \{est, curso \rightarrow prof; prof \rightarrow curso\}$ 
    - Las claves son (est,prof) o (est,curso)
    - Prof no es superclave por lo que viola BCNF.

# Formas Normales

1NF – Esquemas del Modelo Relacional

2NF – Sin Parciales (respecto de claves)

3NF – Sin Parciales ni Transitivas

BCNF –  $X \rightarrow A$  con  $X$  Superclaves