



FACULTAD DE  
**INGENIERIA**  
UNIVERSIDAD NACIONAL DE JUJUY

# Base de Datos I

Dependencia Funcional y Normalización

Mg. Ing. Marcela Tapia

# Objetivos de la clase

- ❖ Aplicar reglas de normalización para :
  - Corregir datos duplicados y anomalías en la base de datos.
  - Evitar la creación y actualización de relaciones y dependencias no deseadas entre datos.
  - Prevenir borrados indeseados de datos.
  - Optimizar el espacio de almacenamiento.

# Normalización

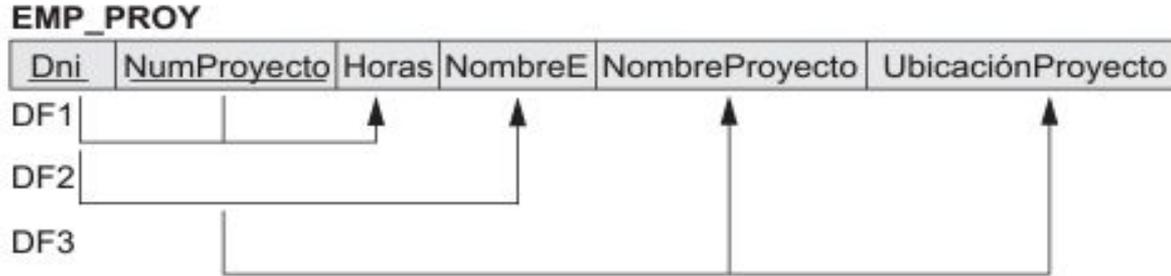
**El proceso de normalización, tal y como fue propuesto en un principio por Codd, hace pasar un esquema de relación por una serie de comprobaciones para certificar que satisface una determinada forma normal. El proceso, que sigue es un método descendente evaluando cada relación contra el criterio de las formas normales y descomponiendo las relaciones según sea necesario, puede considerarse como un diseño relacional por análisis.**

Llevado a la práctica, este proceso comprende la aplicación de una serie de reglas que sirven para ayudar a los diseñadores de bases de datos a desarrollar un esquema que minimice los problemas de lógica.

# Dependencia Funcional

**Definición.** Una dependencia funcional, denotada por  $X \rightarrow Y$ , entre dos conjuntos de atributos  $X$  e  $Y$  que son subconjuntos de  $R$ , especifica una restricción en las posibles tuplas que pueden formar un estado de relación  $r$  de  $R$ .

**Un atributo  $Y$  de una relación depende funcionalmente de otro atributo  $X$  de la relación si a todo valor de  $X$  le corresponde el mismo valor de  $Y$ , en cualquier momento.**



**$Dni \rightarrow NombreEmpleado$**

**$NumProyecto \rightarrow \{NombreProyecto, UbicaciónProyecto\}$**

**$\{Dni, NumeroDpto\} \rightarrow horas$**

# Dependencia Funcional

## PROVEEDOR-ARTÍCULO

Codigo_p	Nombre	Telefono	Codigo_a	Descripcion	Precio
p50	Juan	5576581	A10	tuerca	20
p32	Pedro	3349631	A20	tomillo	30
p50	Juan	5576581	A20	tomillo	32
p32	Pedro	3349631	A10	tuerca	18
p38	Jose	7755633	A49	martillo	45

La tabla PROVEEDOR\_ARTICULO tiene los siguientes campos

Código del Proveedor, Nombre del Proveedor, Teléfono del Proveedor, Código del Artículo, Descripción del Artículo y Precio del Artículo

## Ejemplo Dependencia Funcional

Codigo_p	Nombre	Telefono	Codigo_a	Descripcion	Precio
p50	Juan	5576581	A10	tuerca	20
p32	Pedro	3349631	A20	tomillo	30
p50	Juan	5576581	A20	tomillo	32
p32	Pedro	3349631	A10	tuerca	18
p38	Jose	7755633	A49	martillo	45

Código\_p → Nombre, Teléfono

Codigo\_a → Descripción

Código\_p, Código\_a → Precio

## Ejemplo Dependencia Funcional

Codigo_p	Nombre	Telefono	Codigo_a	Descripcion	Precio
p50	Juan	5576581	A10	tuerca	20
p32	Pedro	3349631	A20	tomillo	30
p50	Juan	5576581	A20	tomillo	32
p32	Pedro	3349631	A10	tuerca	18
p38	Jose	7755633	A49	martillo	45

Codigo_p	Nombre	Telefono	Codigo_a	Descripcion	Precio
p50	Juan	5576581	A10	tuerca	20
p32	Pedro	3349631	A20	tomillo	30
p50	Juan	5576581	A20	tomillo	32
p32	Pedro	3349631	A10	tuerca	18
p38	Jose	7755633	A49	martillo	45

Código\_p → Nombre, Teléfono

**Para todos los Código\_p iguales los registros deben tener el mismo valor de Nombre y Teléfono**

## Ejemplo Dependencia Funcional

Codigo_p	Nombre	Telefono	Codigo_a	Descripcion	Precio
p50	Juan	5576581	A10	tuerca	20
p32	Pedro	3349631	A20	tornillo	30
p50	Juan	5576581	A20	tornillo	32
p32	Pedro	3349631	A10	tuerca	18
p38	Jose	7755633	A49	martillo	45

Codigo_p	Nombre	Telefono	Codigo_a	Descripcion	Precio
p50	Juan	5576581	A10	tuerca	20
p32	Pedro	3349631	A20	tornillo	30
p50	Juan	5576581	A20	tornillo	32
p32	Pedro	3349631	A10	tuerca	18
p38	Jose	7755633	A49	martillo	45

Codigo\_a → Descripción

**Para todos los Código\_a iguales los registros deben tener la misma descripción**

## Ejemplo Dependencia Funcional

Codigo_p	Nombre	Telefono	Codigo_a	Descripcion	Precio
p50	Juan	5576581	A10	tuerca	20
p32	Pedro	3349631	A20	tornillo	30
p50	Juan	5576581	A20	tornillo	32
p32	Pedro	3349631	A10	tuerca	18
p38	Jose	7755633	A49	martillo	45

Codigo\_a  $\nrightarrow$  Precio

**Esta dependencia funcional no se cumple porque dado el mismo codigo\_a los registros no tienen el mismo valor en el atributo precio.**

## Ejemplo Dependencia Funcional

Codigo_p	Nombre	Telefono	Codigo_a	Descripcion	Precio
p50	Juan	5576581	A10	tuerca	20
p32	Pedro	3349631	A20	tomillo	30
p50	Juan	5576581	A20	tomillo	32
p32	Pedro	3349631	A10	tuerca	18
p38	Jose	7755633	A49	martillo	45

Código\_p, Código\_a → Precio

**Como todas las combinaciones de Código\_p y Codigo\_a son distintas la Dependencia Funcional se satisface**

# Dependencia Funcional Completa

Se dice que el atributo Y de la relación R es completo dependiente funcional del atributo X de la relación R, si depende funcionalmente de X y no depende funcionalmente de ningún subconjunto propio de X. ( Es decir, no existe un subconjunto propio Z de los atributos componentes de X tales que Y sea funcionalmente dependiente de Z).

<b>CALIFICACIÓN</b>		
CodAlumno	CodAsignatura	Nota

**DF: CodAlumno, CodAsignatura → Nota**

**Considerando Clave Primaria (CodAlumno, CodAsignatura)**

**@CodAlumno, @CodAsignatura → +Nota**

# Dependencia Funcional Completa

<u>Id_Pais</u>	Nombre	<u>Id_Producto</u>	Cant_Prod
P1	Bolivia	PR1	50
P1	Bolivia	PR2	60
P1	Bolivia	PR3	40
P2	Argentina	PR1	60
P2	Argentina	PR1	60
P3	Ecuador	PR1	30

Considerando como Clave Primaria (Id\_Pais, Id\_Producto)

DF: @Id\_Pais, @ Id\_Producto → + Cant\_Prod

DF: @Id\_Pais → + Nombre (**DEPENDENCIA PARCIAL**)

# Dependencia Funcional Transitiva

Una dependencia funcional  $X \rightarrow Y$  es transitiva cuando un atributo tiene dependencia de otro y éste a su vez de un tercero. Si se tiene los elementos A,B,C y si A es funcionalmente dependiente de B, y B es funcionalmente dependiente de C, entonces A es transitivamente dependiente de C.

<u>DNI</u>	Nombre	Apellido	Cod. Provincia	Provincia
12121349A	Salvador	Velasco	34	Palencia
12121349B	Pedro	Valiente	34	Palencia
3457775G	Ana	Fernández	47	Valladolid
5674378J	Sara	Crespo	47	Valladolid
3456858S	Marina	Serrat	08	Barcelona

DNI  $\rightarrow$  Cod.Provincia

Cod.Provincia  $\rightarrow$  Provincia

Entonces encontramos una dependencia funcional transitiva

DNI  $\rightarrow$  Provincia

## SOLUCIÓN

**Alumnos(#DNI, Nombre, Apellido1, Cod\_Provincia)**

**Provincia(#Cod\_Provincia, Provincia)**

# Primera Forma Normal (1FN)

Una relación está en 1FN si no tiene atributos multivaluados, es decir, si una tupla no toma como valor para un atributo un conjunto de elementos. Es decir si sus atributos contienen valores atómicos.

**“Una relación en la que la intersección de toda fila y columna contiene un valor y solo un valor ”**

PELÍCULAS

Títulos	Directores	Nacionalidades directores	Formatos	Precios	Géneros	Duración	Puntuaciones	Productora ID
El padrino	Francis Ford Coppola	American	HD	30	Gánsteres ,Drama	175	A	1
Toy Story	John Lasseter	American	HD	30	Animación ,Aventura	1	B	2

# Primera Forma Normal (1FN)

Solución 1 - Con redundancia

**PELÍCULAS**

Títulos	Directores	Nacionalidades directores	Formatos	Precios	Géneros	Duración	Puntuaciones	Productora ID
El padrino	Francis Ford Coppola	American	HD	30	Gánsteres	175	A	1
El padrino	Francis Ford Coppola	American	HD	30	Drama	175	A	1
Toy Story	John Lasseter	American	HD	30	Animación	81	B	2
Toy Story	John Lasseter	American	HD	30	Aventura	81	B	2

# Primera Forma Normal (1FN)

## Solución 2

PELÍCULAS

Títulos	Directores	Nacionalidades directores	Formatos	Precios	Duración	Puntuaciones	Productora ID
El padrino	Francis Ford Coppola	American	HD	30	175	A	1
Toy Story	John Lasseter	American	HD	30	81	B	2

GENEROS

ID genero	Nombre genero
1	Gánsteres
2	Drama
3	Animación
4	Aventura

PELÍCULAS - GENEROS

Títulos	ID genero
El padrino	1
El padrino	2
Toy Story	3
Toy Story	4

# Segunda Forma Normal (2FN)

Una relación R está en segunda forma normal si y solo si:

- ❖ Está en Primera Forma Normal
- ❖ Todo atributo que no pertenece a la clave debe depender de la clave en su totalidad y no sólo de una parte, es decir, debe tener Dependencia Funcional Completa

# Segunda Forma Normal (2FN)

The diagram illustrates functional dependencies for the table below. A line originates from the primary key Cod\_trabajador and branches into three arrows pointing to the attributes Nombre, Cod\_area, and Cantidad\_horas, indicating that Cod\_trabajador determines each of these attributes.

<u>Cod_trabajador</u>	Nombre	<u>Cod_area</u>	Nombre_area	Cantidad_horas
001	Juan	FH01	Sistemas	30
002	Pedro	FH02	Alumnos	45
003	Luis	FH03	Administrativa	20
004	María	FH02	Alumnos	36
005	Rosa	FH04	Academica	50
006	Ana	FH01	Sistemas	33

1FN

NO ESTÁ EN 2FN

# Segunda Forma Normal (2FN) - Solución

<u>Cod_trabajador</u>	Nombre
001	Juan
002	Pedro
003	Luis
004	María
005	Rosa
006	Ana

1FN  
2FN

@Cod\_trabajador → + Nombre

<u>Cod_area</u>	Nombre_area
FH01	Sistemas
FH02	Alumnos
FH03	Administrativa
FH04	Academica

1FN  
2FN

@Cod\_area → + Nombre\_area

<u>Cod_trabajador</u>	<u>Cod_area</u>	Cantidad_horas
001	FH01	30
002	FH02	45
003	FH03	20
004	FH02	36
005	FH04	50
006	FH01	33

1FN  
2FN

@Cod\_trabajador + @ Cod\_area → + Cantidad\_horas

# Tercera Forma Normal (3FN)

Una relación está en Tercera Forma Normal si, y solo si:

- ❖ Está en 2FN
- ❖ Todo atributo que no pertenece a la clave no depende de un atributo no clave. (No hay Dependencia Transitiva)

# Tercera Forma Normal (3FN)

El esquema de relación EMP\_DEPT está en 2FN, ya que no existen dependencias no parciales en una clave. Sin embargo, EMP\_DEPT no está en 3FN debido a la dependencia transitiva de DniDirector (y también de NombreDpto) en Dni a través de NúmeroDpto. Podemos normalizar EMP\_DEPT descomponiéndola en los dos esquemas de relación 3FN ED1 y ED2.

