

Dependencias funcionales en BD Relacionales

*Tecnólogo en Informática, sede Paysandú
Bases de Datos 1*

Dependencias funcionales

Temario

- ◆ Directrices de diseño informales
- ◆ Dependencias funcionales
 - Definición
 - Reglas de inferencia
 - Equivalencias
 - Conjuntos mínimos
- ◆ Referencias:
 - *Fundamentals of Database Systems* [E-N], 5ta. Edición, Cap. 10

Dependencias funcionales

Motivación

- ◆ Habíamos visto que:
 - *Esquema de BD relacional* → *esquemas de relación* → *atributos*
- ◆ Podemos encontrarnos con *varios diseños* de solución para el *mismo problema*
- ◆ Vamos a ver parte de la teoría desarrollada con el objetivo de **evaluar** esquemas relacionales encaminados a la **calidad del diseño**

Dependencias funcionales

Motivación

♦ **Objetivo:**

- medir formalmente por qué un conjunto de agrupaciones de atributos en un esquema de relación *es mejor que otro*.

♦ *Directriz* → Instrucción o norma que ha de seguirse en la ejecución de algo.

Dependencias funcionales

Directrices informales

Directrices de diseño informales para los esquemas de relación

- ❖ Antes de tratar la teoría formal, veremos cuatro **medidas informales de calidad** para el diseño de un esquema de relación:
 1. La *semántica* de los atributos.
 2. La reducción de información *redundante* en las tuplas.
 3. La reducción de los *valores NULL* en las tuplas.
 4. Prohibición de la posibilidad de generar *tuplas falsas*.

Directrices informales

Nº1: semántica

Impartir una semántica clara a los atributos de las relaciones

- La **semántica** de una relación hace referencia a la interpretación de los valores de un atributo en una tupla.
 - Relación = conjunto de hechos
- ♦ En general, cuanto más **sencillo** es explicar la semántica de la relación, mejor será el diseño del esquema de relación.

Directrices informales

Nº1: semántica

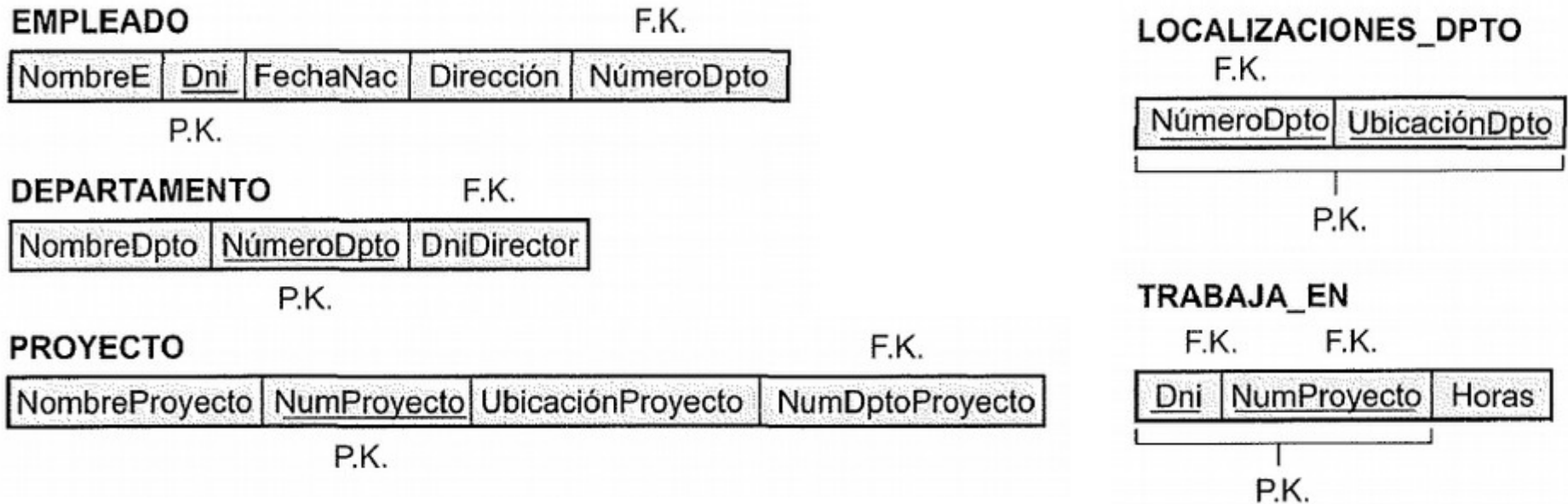
Directriz 1

- ♦ Diseñar un esquema de relación para *que sea fácil explicar su significado*.
- ♦ No combine atributos de varios tipos de entidad y de relación en una única relación.
- ♦ Si la relación está compuesta por una mezcla de múltiples entidades y relaciones, se producirá una ambigüedad semántica y la relación no podrá explicarse con claridad.

Directrices informales

Nº1: semántica

- ◆ Un ejemplo:



- ◆ La facilidad con la que se pueda explicar el significado de los atributos de una relación es una medida informal de lo bien que está diseñada esa relación.

Directrices informales

Nº1: semántica

- ♦ A partir del ejemplo, podría decirse que:
 - Los esquemas tienen una interpretación bien definida y sin ambigüedad.
 - Por consiguiente, todo el esquema de relaciones de la figura podría considerarse como fácil de explicar.
 - Por tanto, es bueno desde el punto de vista de contar con una semántica clara.

Directrices informales

Nº2: redundancias y anomalías

Información redundante en tuplas y anomalías en la actualización

- ❖ Uno de los objetivos de un esquema de diseño es *reducir el espacio de almacenamiento* utilizado por las relaciones.
- ❖ El agrupamiento de atributos en esquemas de relación tiene un efecto significativo sobre el espacio de almacenamiento.

Directrices informales

Nº2: redundancias y anomalías

Directriz 2

- ❖ Diseñar los esquemas de relación de forma que *no se presenten anomalías* de inserción, borrado o actualización en las relaciones.
- ❖ En caso de que aparezca alguna de ellas, anótela claramente y asegúrese de que los programas que actualizan la base de datos operarán correctamente.

Directrices informales

Nº2: redundancias y anomalías

Redundancia

EMP_DEPT

NombreE	Dni	FechaNac	Dirección	NúmeroDpto	NombreDpto	DniDirector
Pérez Pérez, José	123456789	09-01-1965	Eloy I, 98	5	Investigación	333445555
Campos Sastre, Alberto	333445555	08-12-1955	Avda. Ríos, 9	5	Investigación	333445555
Jiménez Celaya, Alicia	999887777	19-07-1968	Gran Vía, 38	4	Administración	987654321
Sainz Oreja, Juana	987654321	20-06-1941	Cerquillas, 67	4	Administración	987654321
Ojeda Ordóñez, Fernando.	666884444	15-09-1962	Portillo, s/n	5	Investigación	333445555
Oliva Avezuela, Aurora	453453453	31-07-1972	Antón, 6	5	Investigación	333445555
Pajares Morera, Luis	987987987	29-03-1969	Enebros, 90	4	Administración	987654321
Ochoa Paredes, Eduardo	888665555	10-11-1937	Las Peñas, 1	1	Sede central	888665555

Directrices informales Nº2: redundancias y anomalías

Pueden haber anomalías en la *inserción*?

Redundancia

EMP_DEPT

NombreE	<u>Dni</u>	FechaNac	Dirección	NúmeroDpto	NombreDpto	DniDirector
Pérez Pérez, José	123456789	09-01-1965	Eloy I, 98	5	Investigación	333445555
Campos Sastre, Alberto	333445555	08-12-1955	Avda. Ríos, 9	5	Investigación	333445555
Jiménez Celaya, Alicia	999887777	19-07-1968	Gran Vía, 38	4	Administración	987654321
Sainz Oreja, Juana	987654321	20-06-1941	Cerquillas, 67	4	Administración	987654321
Ojeda Ordóñez, Fernando.	666884444	15-09-1962	Portillo, s/n	5	Investigación	333445555
Oliva Avezuela, Aurora	453453453	31-07-1972	Antón, 6	5	Investigación	333445555
Pajares Morera, Luis	987987987	29-03-1969	Enebros, 90	4	Administración	987654321
Ochoa Paredes, Eduardo	888665555	10-11-1937	Las Peñas, 1	1	Sede central	888665555

Directrices informales

Nº2: redundancias y anomalías

Y en la actualización?

Redundancia

EMP_DEPT

NombreE	Dni	FechaNac	Dirección	NúmeroDpto	NombreDpto	DniDirector
Pérez Pérez, José	123456789	09-01-1965	Eloy I, 98	5	Investigación	333445555
Campos Sastre, Alberto	333445555	08-12-1955	Avda. Ríos, 9	5	Investigación	333445555
Jiménez Celaya, Alicia	999887777	19-07-1968	Gran Vía, 38	4	Administración	987654321
Sainz Oreja, Juana	987654321	20-06-1941	Cerquillas, 67	4	Administración	987654321
Ojeda Ordóñez, Fernando.	666884444	15-09-1962	Portillo, s/n	5	Investigación	333445555
Oliva Avezuela, Aurora	453453453	31-07-1972	Antón, 6	5	Investigación	333445555
Pajares Morera, Luis	987987987	29-03-1969	Enebros, 90	4	Administración	987654321
Ochoa Paredes, Eduardo	888665555	10-11-1937	Las Peñas, 1	1	Sede central	888665555

Directrices informales Nº2: redundancias y anomalías

Redundancia

EMP_DEPT

NombreE	Dni	FechaNac	Dirección	NúmeroDpto	NombreDpto	DniDirector
Pérez Pérez, José	123456789	09-01-1965	Eloy I, 98	5	Investigación	333445555
Campos Sastre, Alberto	333445555	08-12-1955	Avda. Ríos, 9	5	Investigación	333445555
Jiménez Celaya, Alicia	999887777	19-07-1968	Gran Vía, 38	4	Administración	987654321
Sainz Oreja, Juana	987654321	20-06-1941	Cerquillas, 67	4	Administración	987654321
Ojeda Ordóñez, Fernando.	666884444	15-09-1962	Portillo, s/n	5	Investigación	333445555
Oliva Avezuela, Aurora	453453453	31-07-1972	Antón, 6	5	Investigación	333445555
Pajares Morera, Luis	987987987	29-03-1969	Enebros, 90	4	Administración	987654321
Ochoa Paredes, Eduardo	888665555	10-11-1937	Las Peñas, 1	1	Sede central	888665555

Y con el borrado?

Directrices informales

Nº3: valores *NULL*

Valores NULL en las tuplas

- ❖ En relaciones grandes, si muchos de los atributos no se aplican a todas las tuplas de la relación, nos encontraremos con muchos *NULL* en esas tuplas, lo que puede desperdiciar espacio de almacenamiento.
- ❖ Otro problema con los *NULL* es cómo contabilizarlos cuando se aplican operaciones de agregación como *COUNT* o *SUM*.
- ❖ *NULL* → múltiples interpretaciones.
- ❖ En comparaciones → los resultados serán impredecibles.

Directrices informales

Nº3: valores NULL

Directriz 3

- ◆ Hasta donde sea posible, evite situar en una relación base atributos cuyos valores sean frecuentemente *NULL*.
- ◆ En caso de no poderse evitar, asegúrese de que se aplican sólo en casos excepcionales y no los aplique a la mayor parte de las tuplas de la relación.

Directrices informales

Nº4: tuplas falsas

Generación de tuplas falsas

- ❖ Debido a un mal diseño de las relaciones pueden generarse problemas a la hora de obtener los datos desde varias de ellas.
- ❖ En estos casos, las tuplas que representan información que no es válida reciben el nombre de tuplas falsas.

Directrices informales

N°4: tuplas falsas

Directriz 4

- ❖ Diseñar los esquemas de relación de forma que *puedan concatenarse con condiciones de igualdad* en los atributos que son parejas de clave principal y *foreign key* de forma que se garantice que no se van a generar tuplas falsas.
- ❖ Evite las relaciones que contienen atributos coincidentes que no son combinaciones de *foreign key* y clave principal porque la concatenación de estos atributos puede producir tuplas falsas.

Directrices informales Nº4: tuplas falsas

Personas

<u><i>DniPersona</i></u>	<i>NomPersona</i>	<i>NomProyecto</i>
8888888	Ana	Proyecto 1
5555555	Roberto	Proyecto 2

Proyectos

<u><i>NroProy</i></u>	<i>NomProy</i>
21	Proyecto 1
55	Proyecto 2
32	Proyecto 2

Join Personas-Proyectos

<u><i>DniPersona</i></u>	<i>NomPersona</i>	<i>NomProyecto</i>	<i>NroProy</i>	<i>NomProy</i>
8888888	Ana	Proyecto 1	21	Proyecto 1
5555555	Roberto	Proyecto 2	55	Proyecto 2
5555555	Roberto	Proyecto 2	32	Proyecto 2

Directrices informales Nº4: tuplas falsas

Personas

<u>DniPersona</u>	NomPersona	NomProyecto
8888888	Ana	Proyecto 1
5555555	Roberto	Proyecto 2

Proyectos

<u>NroProy</u>	NomProy
21	Proyecto 1
55	Proyecto 2
32	Proyecto 2

Join Personas-Proyectos

<u>DniPersona</u>	NomPersona	NomProyecto	<u>NroProy</u>	NomProy
8888888	Ana	Proyecto 1	21	Proyecto 1
5555555	Roberto	Proyecto 2	55	Proyecto 2
5555555	Roberto	Proyecto 2	32	Proyecto 2

¿Ambas son correctas?

Directrices informales

Resumen

- ◆ Hemos visto **situaciones** que provocan esquemas de relación problemáticos, que implica trabajo redundante, desaprovechamiento de almacenamiento o generación de datos inválidos.
- ◆ Tratamos unas directrices informales para definir un **buen diseño relacional**.
- ◆ A continuación se presentarán conceptos y teorías formales que pueden utilizarse para definir de forma más precisa la *idoneidad* y la *mala calidad* de un esquema de relación individual.

Dependencias funcionales

Dependencias funcionales

Concepto

- ♦ Una dependencia funcional (*DF*) es una **restricción** que se establece entre *dos conjuntos de atributos* de la base de datos.
- ♦ **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 .
 - La restricción dice que dos tuplas $t1$ y $t2$ en r tales que cumplen que $t1[X] = t2[X]$, deben cumplir también que $t1[Y] = t2[Y]$.

Dependencias funcionales

Concepto

- ♦ Por tanto:
 - X **determina funcionalmente** Y si para toda instancia r del esquema de relación R , no es posible que r tenga dos tuplas que coincidan en los atributos de X y no lo hagan en los atributos de Y .
- ♦ Observe lo siguiente:
 - Si X es una **clave candidata** de R , se cumple que $X \rightarrow Y$ para cualquier subconjunto de atributos Y de R (la restricción de clave implica que dos tuplas en cualquier estado legal $r(R)$ no tendrán el mismo valor de X).
 - Si $X \rightarrow Y$ en R , esto **no supone** que $Y \rightarrow X$ en R .

Dependencias funcionales

Concepto

- ❖ Una dependencia funcional es una **propiedad de la semántica** o significado de los atributos.
- ❖ Las dependencias funcionales **deben mantenerse** en todos los estados de relación r de R .
- ❖ Los estados de relación $r(R)$ que satisfacen la restricción de DF reciben el nombre de **estados de relación legales**.

Dependencias funcionales

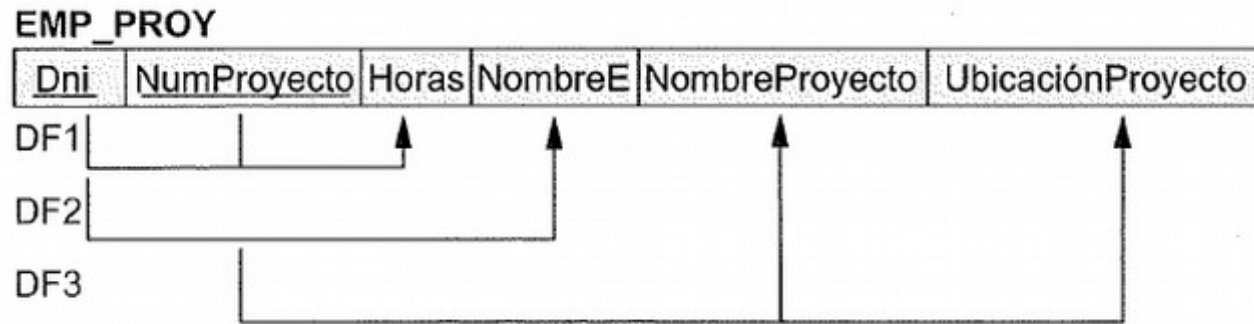
Concepto

- ◆ Por tanto, el **uso fundamental** de las DFs es *describir* más en profundidad un esquema de relación R especificando restricciones de sus atributos que siempre deben cumplirse.
- ◆ Ciertas DFs pueden especificarse sin hacer referencia a una relación específica.
- ◆ Por ejemplo:
 - $\{Provincia, NumPermisoConducir\} \rightarrow Dni$ debe mantenerse para cualquier adulto que viva en España.

Dependencias funcionales

Concepto

- ◆ **Ejemplo:** considerar el esquema relación *EMP_PROY* y sus *DFs*



- ◆ $\{Dni, NumProyecto\} \rightarrow Horas$
- ◆ $Dni \rightarrow NombreE$
- ◆ $NumProyecto \rightarrow \{NombreProyecto, UbicaciónProyecto\}$

Dependencias funcionales

Concepto

- ♦ Una *DF* es un **propiedad del esquema**, no de la instancia
- ♦ No puede ser inferida automáticamente a partir de un estado de relación *r*, sino que alguien que conozca la semántica de los atributos de *R* **debe definirla explícitamente**.

Figura 10.7. Un estado de relación IMPARTIR con una *posible* dependencia funcional TEXTO → CURSO. Sin embargo, PROFESOR → CURSO no es posible.

IMPARTIR

Profesor	Curso	Texto
Smith	Estructuras de datos	Bartram
Smith	Administración de datos	Martin
Hall	Compiladores	Hoffman
Brown	Estructuras de datos	Horowitz

Dependencias funcionales

Ejemplo

- ♦ Una empresa de alquiler de vehículos desea implementar una base de datos con la información de su negocio. Se tienen vehículos identificados por su número de matrícula, y de los que se conoce su marca, color, modelo y año.
- ♦ También se tienen clientes identificados por su número de cédula de identidad, y de los que se conoce su nombre, dirección y teléfono. Un contrato de alquiler de vehículo está identificado por un número de contrato y se realiza en una fecha dada entre un cliente y un vehículo, registrándose el periodo de alquiler en días y el precio del servicio.
- ♦ Se considera que en una misma fecha no se puede alquilar más de una vez el mismo vehículo al mismo cliente en la misma fecha.

Dependencias funcionales

Ejemplo

- ♦ Una empresa de alquiler de vehículos desea implementar una base de datos con la información de su negocio. Se tienen **vehículos** identificados por su número de **matrícula**, y de los que se conoce su **marca**, **color**, **modelo** y **año**.
- ♦ También se tienen **clientes** identificados por su número de **cédula** de identidad, y de los que se conoce su **nombre**, **dirección** y **teléfono**. Un **contrato** de alquiler de vehículo está identificado por un **número de contrato** y se realiza en una **fecha** dada entre un **cliente** y un **vehículo**, registrándose el **periodo** de alquiler en días y el **precio** del servicio.
- ♦ Se considera que **en una misma fecha no se puede alquilar más de una vez el mismo vehículo al mismo cliente** en la misma fecha.

Dependencias funcionales

Ejemplo

◆ **Entonces obtenemos:**

matrícula → {*marca, color, modelo, año*}

cédula → {*nombre, dirección, teléfono*}

nroContrato → {*fecha, cédula, matrícula, período, precio*}

{*fecha, cédula, matrícula*} → *nroContrato*

Dependencias funcionales

Inferencia, deducción

- ◆ Decimos que:
 - F : conjunto de DFs especificadas en un esquema de relación R .
- ◆ **Habitualmente** se especifican las DFs que son *semánticamente obvias*.
- ◆ Sin embargo, es habitual que **muchas otras** DFs se encuentren en todas las instancias de relación legales entre los conjuntos de atributos que pueden derivarse y satisfacen las dependencias de F .

Dependencias funcionales

Inferencia, deducción

- ◆ Entonces, decimos que esas otras dependencias pueden ***inferirse*** o ***deducirse*** de las *DF* de *F*.
- ◆ En la vida real, es **imposible especificar todas** las dependencias funcionales posibles para una situación concreta.

Dependencias funcionales

Inferencia, deducción

♦ **Por ejemplo:**

- Si cada departamento tiene un director, de manera que $NroDpto$ determina de forma única $DniDirector$ ($NroDpto \rightarrow DniDirector$)
- y un director tiene un único número de teléfono $TelDirector$ ($DniDirector \rightarrow TeléfonoDirector$)

♦ **Entonces:**

- Ambas dependencias juntas suponen que $NroDpto \rightarrow TeléfonoDirector$.
- Esto es una DF inferida y no tiene que declararse *explícitamente*.

Dependencias funcionales

Clausura (F^+)

- ◆ Por tanto, formalmente es útil definir un concepto llamado *clausura* (*closure*) que incluye *todas las posibles* dependencias que pueden **inferirse** de un conjunto F dado
- ◆ **Clausura (F^+):**
 - Es el conjunto de todas las dependencias que incluyen F , **junto con** las dependencias que pueden inferirse de F , reciben el nombre de clausuras de F ; está designada mediante F^+ .

Nota: Tener en cuenta que las DF en F^+ deben cumplirse *para todos* los estados $r(R)$ legales donde se cumpla F

Dependencias funcionales

Reglas de inferencia

- ❖ Para determinar una **manera sistemática** de inferir dependencias, debemos descubrir un conjunto de *reglas de inferencia* que puedan usarse para *deducir nuevas dependencias* a partir de un conjunto de dependencias concreto.
- ❖ Usamos la notación $F \models X \rightarrow Y$ para indicar que la dependencia funcional $X \rightarrow Y$ **se infiere** del conjunto de dependencias funcionales F .

Dependencias funcionales

Reglas de inferencia

- ◆ **Reglas:** (siendo W, X, Y, Z conjuntos de atributos)
 - (RI1) **reflexiva:** Si $X \supseteq Y$, entonces $X \rightarrow Y$
 - (RI2) **de aumento:** $\{X \rightarrow Y\} \models XZ \rightarrow YZ$
 - (RI3) **transitiva:** $\{X \rightarrow Y, Y \rightarrow Z\} \models X \rightarrow Z$
 - (RI4) **descomposición:** $\{X \rightarrow YZ\} \models X \rightarrow Y$
 - (RI5) **unión:** $\{X \rightarrow Y, X \rightarrow Z\} \models X \rightarrow YZ$
 - (RI6) **pseudotransitiva:** $\{X \rightarrow Y, WY \rightarrow Z\} \models WX \rightarrow Z$
- ◆ **Reglas de Armstrong:** RI1 a RI3
 - **Minimales:** Las demás se pueden derivar a partir de estas tres.

Dependencias funcionales

Reglas de inferencia

♦ **Nota:**

- Aunque $X \rightarrow A$ y $X \rightarrow B$ implican $X \rightarrow AB$ por la regla de unión (RI5),
- $X \rightarrow A$ e $Y \rightarrow B$ **no implican** que $XY \rightarrow AB$.
- Además, $XY \rightarrow A$ **no implica** necesariamente ni $X \rightarrow A$ ni $Y \rightarrow A$.

Dependencias funcionales

Clausura de X bajo F (X^+)

- ◆ **F** : conjunto de *DFs* sobre un esquema
- ◆ **X** : conjunto de atributos que aparece en la *parte izquierda* de alguna dependencia funcional de F
- ◆ **Definición:**
 - La clausura de X bajo F es el conjunto de atributos *determinados funcionalmente* por X , basados en F ; está designada por X^+

Dependencias funcionales

Clausura de X bajo F (X^+)

♦ Algoritmo para determinar X^+ bajo F

$X^+ := X$

Asigna todos los atributos de X

repetir

antigua $X^+ := X^+$;

para cada $df Y \rightarrow Z$ *en* F *hacer*

si $Y \subseteq X^+$ *entonces* $X^+ := X^+ \cup Z$;

Agregar atributos a X^+

hasta que (*antigua* $X^+ = X^+$);

No hay más cambios en X^+

Dependencias funcionales

Clausura de X bajo F (X^+)

◆ Ejemplo:

- *EMP_PROY* (*NSS*, *NumProy*, *Horas*, *NomEmp*, *NomProy*, *LugarProy*)
- $F = \{NSS \rightarrow NomEmp;$
 $NumProy \rightarrow NomProy, LugarProy;$
 $NSS, NumProy \rightarrow Horas\}$

◆ $\{NSS\}^+ = \{NSS, NomEmp\}$

◆ $\{NumProy\}^+ = \{NumProy, NomProy, LugarProy\}$

◆ $\{NSS, NumProy\}^+ = \{NSS, NumProy, NomEmp, NomProy, LugarProy, Horas\}$

Dependencias funcionales

Equivalencias de conjuntos de DFs

◆ Definición:

- Dos conjuntos de dfs E y F **son equivalentes** sii $E^+ = F^+$.
- Entonces:
 - Todas las dfs en E se pueden **inferir** de F y todas las de F se pueden **inferir** de E .
 - E **cubre** a F y F **cubre** a E .

◆ ¿Cómo determinamos si F cubre a E ?

- Para cada df $X \rightarrow Y \in E$, calculamos X^+ *respecto a* F y verificamos que X^+ incluya los atributos en Y .

Dependencias funcionales

Equivalencias de conjuntos de DFs

◆ Ejemplo:

- $F = \{AB \rightarrow C, B \rightarrow D, D \rightarrow GC, CG \rightarrow H\}$
- $F1 = \{D \rightarrow H, B \rightarrow C, AD \rightarrow GH\}$
 - $F1$ cubre a F ?
 - F cubre a $F1$?
 - F es equivalente a $F1$?

Dependencias funcionales

Equivalencias de conjuntos de DFs

♦ Ejemplo:

- $F = \{AB \rightarrow C, B \rightarrow D, D \rightarrow GC, CG \rightarrow H\}$
- $F1 = \{D \rightarrow H, B \rightarrow C, AD \rightarrow GH\}$
- $F2 = \{B \rightarrow D, D \rightarrow G, D \rightarrow C, CG \rightarrow H\}$
 - Qué pasa entre $F2$ y F ?
 - Qué pasa entre $F1$ y $F2$?

Dependencias funcionales

Conjuntos mínimos de DFs

- ♦ Una **cobertura mínima** de un conjunto de dfs E , es un conjunto de dfs F que satisface la propiedad que cada dependencia de E **está en** F^+ de F .
- ♦ Además, esta propiedad **se pierde** si se *elimina* cualquier df de F .
- ♦ F *no debe* tener **redundancias**, y las dependencias en F están en una *forma estándar*.

Dependencias funcionales

Conjuntos mínimos de DFs

♦ F es **minimal** sii:

1. **Toda** df en F **tiene** *un solo* atributo a la derecha
2. **No** podemos reemplazar **ninguna** df $X \rightarrow A \in F$ por una df $Y \rightarrow A$, donde Y es subconjunto de X , y *seguir teniendo* un conjunto de dfs **equivalente** a F
3. **No** podemos eliminar **ninguna** dependencia de F y *seguir teniendo* un conjunto de dfs **equivalente** a F

Dependencias funcionales

Conjuntos mínimos de DFs

♦ Algoritmo para localizar cobertura mínima F para E

Donde: E, F : conj. de dfs; X, Y : conj. de atributos; A, B : atributos individuales

1. Establecer $F := E$

Reemplazar dependencias

2. Reemplazar cada df $X \rightarrow \{A1, A2, \dots, An\}$ en F por las n dfs $X \rightarrow A1, X \rightarrow A2, \dots, X \rightarrow An$

3. Por cada df $X \rightarrow A$ en F

por cada atributo B que sea un elemento de X

si $\{(X-B)^+ \text{ respecto a } F, \text{ contiene a } A\}$

entonces reemplazar $X \rightarrow A$ por $(X - \{B\}) \rightarrow A$ en F

Atributos redundantes

4. Por cada df $X \rightarrow A$ sobrante en F

si $\{F - \{X \rightarrow A\}\}$ es equivalente a F ,

entonces eliminar $X \rightarrow A$ de F .

Dependencias redundantes

Dependencias funcionales

Conjuntos mínimos de DFs

♦ **Ejemplo:** Dado el conjunto de dfs $E = \{B \rightarrow A, D \rightarrow A, AB \rightarrow D\}$, hallar el cubrimiento minimal.

– Hay df en E con varios **atributos a la derecha?** Ninguna

Reemplazar dependencias

– Hay **atributos redundantes?** En $AB \rightarrow D$:

- $\{A\}^+ = \{A\}$
- $\{B\}^+ = \{B, A, D\}$
- Entonces obtenemos $E1 = \{B \rightarrow A, D \rightarrow A, B \rightarrow D\}$

Sin atributos redundantes

– Hay **dfs redundantes?** En $E1$

- En $E1 - \{B \rightarrow A\}$, calculo $\{B\}^+ = \{B, D, A\}$
- Por lo tanto, $B \rightarrow A$ es redundante en $E1$ y puede eliminarse
- Entonces obtenemos $E2 = \{D \rightarrow A, B \rightarrow D\}$

Sin dfs redundantes

– Entonces: $\{D \rightarrow A, B \rightarrow D\}$ es **cubrimiento minimal de E**