Normalización - Equivalencia

- **Cubrimiento.** Dados E y F conjuntos de DFs, F **cubre** a E si $(\forall df \in E) df \in F^+$
- **Equivalencia.** Dados E y F conjuntos de DFs, F y E son **equivalentes** si $F^+ = E^+$, es decir, si F **cubre** a E y E **cubre** a F
- **Ejercicio.** Decir si los siguientes conjuntos de DFs son equivalentes
  - $F = \{A \to C, AC \to D, E \to AD, E \to H\}$
  - $G = \{A \to CD, E \to AH\}$
- **Procedimiento.** Para determinar si F **cubre** a G, calcular, para cada DF $X \to Y$ de G, X^+ con respecto a F . Luego verificar si este X^+ incluye los atributos en Y. Similar razonamiento para verificar si G **cubre** a F

Normalización - Conjunto Minimal de DFs

- $\rightarrow$ Se explico cómo expandir F a F^+
- $\leftarrow$ Se quiere ver el camino inverso, reducir F a su expresión minimal
- **Atributo Extraño.** Atributo que puede ser removido sin alterar la clausura del conjunto de DFs.
- **Formalmente.** Sea $X \to A$ en F, $Y \subset X$ es **extraño** si F implica lógicamente $(F - \{X \to A\} \cup \{(X - Y) \to A\}$
- **Características de un Conjunto de DFs para ser minimal**
  1. Cada DF de F debe poseer un solo atributo en su lado derecho
  2. No es posible reemplazar niguna DF $X \to A$ de F por $Y \to A$, siendo $Y \subset X$, y seguir teniendo un conjunto de DFs equivalente a F
  3. No es posible remover niguna DF de F y seguir teniendo un conjunto de DFs equivalente a F
- **Intuitivamente.** F *minimal* es un conjunto *canónico* y *sin redundancia*
- **Cubrimiento minimal.** Un **cubrimiento minimal** de F es un conjunto minimal de DFs (en forma canónica y sin redundancia) que es equivalente a F.
- **Existencia.** Siempre es posible hallar al menos un **cubrimiento minimal** F para cualquier conjunto de DFs E usando el siguiente algoritmo

Normalización - Conjunto Minimal de DFs (Cont.)

Algoritmo Nro. 2 Búsqueda de un cubrimiento minimal F para un conjunto de DFs E

Entrada: Conjunto de DFs E

0. $F := E$
1. **Reemplazar** *cada DF* $X \to \{A_1, A_2, \ldots, A_n\}$ *en* F **por** *n DFs* $X \to A_1, X \to A_2, \ldots, X \to A_n$

/*Traslada a todas las DFs a una forma canónica para los pasos subsiguientes*/

2. **Para cada** *DF* $X \to A$ **en** F
   - **Para cada** *atributo B que es un* **elemento de** X
     - **Si** $\{F - \{X \to A\}\} \cup \{(X - \{B\}) \to A\}$ *es equivalente a* F
       - **Entonces reemplazar** $X \to A$ **por** $(X - \{B\}) \to A$

/*Remueve al atributo extraño B del lado izquierdo de X siempre que es posible*/

3. **Para cada** *DF* $X \to A$ **en** F
   - **Si** $F - \{X \to A\}$ *es equivalente a* F
     - **Remover** $X \to A$ *de* F

/*Remueve las DF redundantes siempre que es posible*/

Normalización - Conjunto Minimal de DFs (Cont.)

- **Ejemplo 1.** Sea un conjunto de DFs $E = \{B \to A, D \to A, AB \to D\}$. Encontrar el cubrimiento minimal de E denominado F
  - **Paso (1)** Todas las DFs de E están en forma canónica. No es necesario hacer ningún cambio
  - **Paso (2)** Hay que determinar si $AB \to D$ posee algún atributo extraño en su lado izquierdo. Esto es, si puede ser reemplazado por $A \to D$ o $B \to D$
    - Aplicando RI2 a $B \to A$, incrementándolo con B, se obtiene $BB \to AB$ que equivale a (i) $B \to AB$; Adicionalmente se tiene la DF (ii) $AB \to D$
    - Aplicando la RI3 (transitiva) sobre (i) y (ii), se obtiene $B \to D$. Así, $AB \to D$ puede ser reemplazada por $B \to D$
    - El conjunto original E puede ser reemplazado por otro equivalente $E' = \{B \to A, D \to A, B \to D\}$
    - No es posible otra reducción ya que todos los lados izquierdos poseen un solo atributo
  - **Paso (3)** Usando RI3 (transitiva) sobre $B \to D$ y $D \to A$, se infiere $B \to A$. Por lo tanto $B \to A$ es redundante y puede ser eliminada de E'
  - **Cubrimiento minimal de E.** $F = \{B \to D, D \to A\}$

Normalización - Lossless Join (Cont.)

- **Algoritmo Nro. 4** Chequeo de propiedad Lossless Join

Entrada: R, descomposición $D=\{R_1,R_2,\dots,R_m\}$ de R y un conjunto de DFs F

1. Crear una matriz S con una fila i por cada R_i en D, y una columna j por cada atributo A_j en R
2. **Para todo** i,j **asignar** $S(i,j)=b_{ij}$ /*cada b_{ij} es un elemento distinto de la matriz*/
3. **Para cada** i,j
   Si $A_j \in R_i$ **entonces** $S(i,j)=a_j$ /*distingue a elementos que pertenecen a la relación R_i*/
4. **Repetir hasta** *que un loop completo no genere cambios en S*
   Para cada $X \rightarrow Y$ *en F*
     Para todas *las filas fs en S que <u>tienen los mismos valores</u> en los atributos de X*
       Hacer *que los atributos en fs para cada columna y de Y tengan el mismo valor de la siguiente manera*
         Si *alguna de las fs en y tiene un simbolo a* **entonces asignarlo** *al resto de las fs en y*
         Sino *elegir arbitrariamente un simbolo b de fs en y y* **asignarlo** *al resto de las fs en y*
5. **Si** *alguna fila de S posee la totalidad de elementos a* **entonces** *es* **lossless join**, **caso contrario** *no lo es*

Normalización - Lossless Join (Cont.)

- **Ejemplo 1.** Sean
  - $R=\{E\_CUIL, E\_Nombre, P\_Número, P\_Nombre, P\_Ubicación, Horas\}$
  - $R_1=EMP\_UBICACION=\{E\_Nombre, P\_Ubicación\}$
  - $R_2=EMP\_PROY1=\{E\_CUIL, P\_Número, P\_Nombre, P\_Ubicación, Horas\}$
  - $D=\{R_1, R_2\}$
  - $F=\{$
    $E\_CUIL \rightarrow E\_Nombre;$
    $P\_Número \rightarrow \{P\_Nombre; P\_Ubicación\};$
    $\{E\_CUIL, P\_Número\} \rightarrow Horas;\}$
  - **Paso 1.**

	E_CUIL	E_Nombre	P_Número	P_Nombre	P_Ubicación	Horas
R_1						
R_2						

  - **Paso 2.**

	E_CUIL	E_Nombre	P_Número	P_Nombre	P_Ubicación	Horas
R_1	b_{11}	b_{12}	b_{13}	b_{14}	b_{15}	b_{16}
R_2	b_{21}	b_{22}	b_{23}	b_{24}	b_{25}	b_{26}

  - **Paso 3.**

	E_CUIL	E_Nombre	P_Número	P_Nombre	P_Ubicación	Horas
R_1	b_{11}	a_2	b_{13}	b_{14}	a_5	b_{16}
R_2	a_1	b_{22}	a_3	a_4	a_5	a_6

  - **Paso 4.** No modifica ningún símbolo b en a
  - **Paso 5.** No hay ninguna fila en S que posea a en la totalidad de valores, por lo tanto la descomposición no es lossless join

Normalización - Lossless Join (Cont.)

- **Ejemplo 1.** Sean
  - $R=\{E\_CUIL, E\_Nombre, P\_Número, P\_Nombre, P\_Ubicación, Horas\}$
  - $R_1=EMP=\{E\_CUIL, E\_Nombre\}$
  - $R_2=PROY=\{P\_Número, P\_Nombre, P\_Ubicación\}$
  - $R_3=TRABAJA\_EN=\{E\_CUIL, P\_Número, Horas\}$
  - $D=\{R_1, R_2, R_3\}$
  - $F=\{\ E\_CUIL \rightarrow E\_Nombre;\ P\_Número \rightarrow \{P\_Nombre; P\_Ubicación\};$
    $\{E\_CUIL, P\_Número\} \rightarrow Horas;\}$
  - **Paso 1.**

	E_CUIL	E_Nombre	P_Número	P_Nombre	P_Ubicación	Horas
R_1						
R_2						
R_3						

  - **Paso 2.**

	E_CUIL	E_Nombre	P_Número	P_Nombre	P_Ubicación	Horas
R_1	b_{11}	b_{12}	b_{13}	b_{14}	b_{15}	b_{16}
R_2	b_{21}	b_{22}	b_{23}	b_{24}	b_{25}	b_{26}
R_3	b_{31}	b_{32}	b_{33}	b_{34}	b_{35}	b_{36}

  - **Paso 3.**

	E_CUIL	E_Nombre	P_Número	P_Nombre	P_Ubicación	Horas
R_1	a_1	a_2	b_{13}	b_{14}	b_{15}	b_{16}
R_2	b_{21}	b_{22}	a_3	a_4	a_5	b_{26}
R_3	a_1	b_{32}	a_3	b_{34}	b_{35}	a_6

Normalización - Lossless Join (Cont.)

- **Ejemplo 1.** Sean
  - $R=\{E\_CUIL, E\_Nombre, P\_Número, P\_Nombre, P\_Ubicación, Horas\}$
  - $R_1=EMP=\{E\_CUIL, E\_Nombre\}$
  - $R_2=PROY=\{P\_Número, P\_Nombre, P\_Ubicación\}$
  - $R_3=TRABAJA\_EN=\{E\_CUIL, P\_Número, Horas\}$
  - $D=\{R_1, R_2, R_3\}$
  - $F=\{\ E\_CUIL \rightarrow E\_Nombre;\ P\_Número \rightarrow \{P\_Nombre; P\_Ubicación\};$
    $\{E\_CUIL, P\_Número\} \rightarrow Horas;\}$
  - **Paso 4.** $E\_CUIL \rightarrow E\_Nombre$

	E_CUIL	E_Nombre	P_Número	P_Nombre	P_Ubicación	Horas
R_1	a_1	a_2	b_{13}	b_{14}	b_{15}	b_{16}
R_2	b_{21}	b_{22}	a_3	a_4	a_5	b_{26}
R_3	a_1	$\cancel{b_{32}}$ a_2	a_3	b_{34}	b_{35}	a_6

  - **Paso 4.** $P\_Número \rightarrow \{P\_Nombre; P\_Ubicación\}$

	E_CUIL	E_Nombre	P_Número	P_Nombre	P_Ubicación	Horas
R_1	a_1	a_2	b_{13}	b_{14}	b_{15}	b_{16}
R_2	b_{21}	b_{22}	a_3	a_4	a_5	b_{26}
R_3	a_1	$\cancel{b_{32}}$ a_2	a_3	$\cancel{b_{34}}$ a_4	$\cancel{b_{35}}$ a_5	a_6

  - **Paso 4.** $\{E\_CUIL, P\_Número\} \rightarrow Horas$ no produce cambios en S
  - **Paso 4.** Nueva vuelta sobre TODAS las DFs F no produce cambios en S
  - **Paso 5.** Última fila de S posee la totalidad de sus valores en a, por lo tanto la descomposición es lossless join

Normalización - Lossless Join - Descomposiciones sucesivas

- **Recapitulando.** En ejemplos previos utilizamos descomposiciones sucesivas al pasar a R a 2FN y luego a 3FN

Afirmación Nro. 2

Si se cumplen las siguientes condiciones:

- Una descomposición $D=\{R_1, R_2, \ldots, R_m\}$ de R cumple la propiedad de lossless join con respecto a F de R
- Una descomposición $D_i=\{Q_1, Q_2, \ldots, Q_k\}$ de R_i cumple la propiedad de lossless join con respecto a la proyección de F sobre R_i

Entonces la descomposición $D_2=\{R_1, R_2, \ldots R_{i-1}, Q_1, Q_2, \ldots, Q_k, R_{i+1}, \ldots, R_m\}$ de R cumple con la propiedad lossless join con respecto a F de R

Normalización - Algoritmos Diseño

- **Algortimo D1.** Descompone relación universal R cumpliendo:
  - 3FN
  - Preservación de DFs
  - Lossless Join
- **Algortimo D2.** Descompone relación universal R cumpliendo:
  - BCFN
  - Lossless Join
- No es posible diseñar algoritmo que produzca una descomposición en BCFN con preservación DFs y Lossless Join

Normalización - Algoritmos Diseño 1 - 3FN

- **Algoritmo Nro. D1** Descomposición en 3FN

Entrada: R universal y un conjunto de DFs F sobre R

1. Hallar el cubrimiento minimal G de F (utilizar algoritmo ya dado)
2. **Para cada** *lado izquierdo X de cada DF que aparece en G*
   Crear *una relación en D con atributos* $\{X \cup \{A_1\} \cup \{A_2\} \cup \ldots \cup \{A_k\}\}$
   siendo $X \rightarrow A_1, X \rightarrow A_2, \ldots, X \rightarrow A_k$ las únicas dependencias
   en G con X como lado izquierdo (X es la clave de esta relación)
3. **Si** *ninguna relación en D contiene una clave de R*
   entonces *crear una relación adicional en D que contenga*
   atributos que formen una clave de R (se puede utilizar algoritmo ya dado)
4. **Eliminar** *relaciones redundantes de D. Una relación R de D es redundante si R es una proyección de otra relación S de D*

Normalización - Algoritmos Diseño 1 - 3FN (Cont.)

- **Ejemplo 1.**
  - $U=\{\text{E\_CUIL}, \text{P\_Número}, \text{E\_Salario}, \text{E\_Teléfono}, \text{D\_Número}, \text{P\_Nombre}, \text{P\_Ubicación}\}$
  - $F=\{$ FD1: $\text{E\_CUIL} \rightarrow \{\text{E\_Salario}, \text{E\_Teléfono}, \text{D\_Número}\}$,
    FD2: $\text{P\_Número} \rightarrow \{\text{P\_Nombre}, \text{P\_Ubicación}\}$,
    FD3: $\{\text{E\_CUIL}, \text{P\_Número}\} \rightarrow \{\text{E\_Salario}, \text{E\_Teléfono}, \text{D\_Número}, \text{P\_Nombre}, \text{P\_Ubicación}\}\}$
  - $\{\text{E\_CUIL}, \text{P\_Número}\}$ representa una clave de la relación U (por $FD3$)
  - **Paso 1.** Aplicando algoritmo de minimal cover, en su paso 3 se observa
    - P_Número es atributo extraño en $\{\text{E\_CUIL}, \text{P\_Número}\} \rightarrow \{\text{E\_Salario}, \text{E\_Teléfono}, \text{D\_Número}\}$
    - E_CUIL es atributo extraño en $\{\text{E\_CUIL}, \text{P\_Número}\} \rightarrow \{\text{P\_Nombre}, \text{P\_Ubicación}\}$
    - Así, cubrimiento minimal = FD1 y FD2 (FD3 es redundante). Agrupando atributos con mismo lado izq. en una sola DF:

    Cubrimiento minimal $G=\{\text{E\_CUIL} \rightarrow \{\text{E\_Salario}, \text{E\_Teléfono}, \text{D\_Número}\}$,
    $\text{P\_Número} \rightarrow \{\text{P\_Nombre}, \text{P\_Ubicación}\}\}$
  - **Paso 2.** Producir relaciones R_1 y R_2
    - $R_1=(\underline{\text{E\_CUIL}}, \text{E\_Salario}, \text{E\_Teléfono}, \text{D\_Número})$
    - $R_2=(\underline{\text{P\_Número}}, \text{P\_Nombre}, \text{P\_Ubicación})$
  - **Paso 3.** Generar R_3 adicional con clave de U. Obteniendo finalmente:
    - $R_1=(\underline{\text{E\_CUIL}}, \text{E\_Salario}, \text{E\_Teléfono}, \text{D\_Número})$
    - $R_2=(\underline{\text{P\_Número}}, \text{P\_Nombre}, \text{P\_Ubicación})$
    - $R_3=(\underline{\text{E\_CUIL}, \text{P\_Número}})$

Normalización - Algoritmos Diseño 2 - BCFN

- **Algoritmo Nro. D2** Descomposición en BCFN

  Entrada: R universal y un conjunto de DFs F sobre R

  1. $D:=\{R\}$
  2. **Mientras** $(\exists Q \in D)$ *Q no cumple BCFN* {
     Seleccionar $Q \in D$ *que no cumple BCFN*;
     Encontrar *DF* $X \rightarrow Y$ *en Q que no cumple con BCFN*;
     Reemplazar *Q en D por la siguientes dos relaciones*: $(Q - Y)$ *y* $(X \cup Y)$;
     };

- En base a la propiedad NJB (descomposición binaria) y a la Afirmación Nro. 2 D cumple con la propiedad lossless join

Normalización - Algoritmos Diseño 2 - BCFN

- **Ejemplo.**
  - $R=\{\underline{Estudiante}, \underline{Materia}, Instructor\}$
  - $F=\{$ $FD1: \{Estudiante, Materia\} \rightarrow Instructor,$
    $FD2: Instructor \rightarrow Materia\}$
- Aplicando el algoritmo se obtiene
  - $R_1=(\underline{Estudiante}, \underline{Instructor})$
  - $R_2=(\underline{Instructor}, Materia)$

Importante

La teoría de lossless join se basa en la asunción de que no existen valores NULL en los atributos de JOIN

Normalización - Bibliografía

- Capítulo 15 (hasta 15.3 inclusive) Elmasri/Navathe - Fundamentals of Database Systems, 7th Ed., Pearson, 2015.