

Solución Segundo Parcial de Fundamentos de Bases de Datos

Noviembre 2013

Ejercicio 1 (12 pts)

Considere el esquema de relación $R(A,B,C,D,E)$ y el conjunto de dependencias funcionales $F=\{ A \rightarrow B, C \rightarrow E, B \rightarrow D \}$. Considere además la siguiente instancia válida de R , donde subíndice diferentes sobre letras iguales representan valores diferentes (Ej: $a_1 \neq a_2$).

A	B	C	D	E
a_1	b_1	c_1	d_1	e_1
a_1	b_1	c_2	d_1	e_2
a_2	b_2	c_2	d_2	e_3
a_2	b_2	c_3	d_3	e_4

1. Indicar cuáles de las dependencias funcionales de F se cumplen y cuáles no se cumplen en el esquema de relación R . Justifique su respuesta.

SOLUCIÓN:

Dependencia	Se cumple?
$A \rightarrow B$	No es posible afirmarlo. La instancia dada satisface la dependencia pero no podemos afirmarlo para todas las instancias como lo requiere la definición de dependencia funcional.
$C \rightarrow E$	No, ya que para las tuplas 2 y 3 los valores del atributo C son iguales (c_2) pero sus respectivos valores del atributo E son diferentes (e_2, e_3).
$B \rightarrow D$	No, ya que para las tuplas 3 y 4 los valores del atributo B son iguales (b_2) pero sus respectivos valores del atributo D son diferentes (d_2, d_3).

2. Indicar si la dependencia multivaluada embebida $A \twoheadrightarrow D \mid BC$ se cumple en R . Justifique su respuesta.

SOLUCIÓN:

Por definición, la dependencia embebida anterior se cumplirá en R si para el subesquema (A, B, C, D) se cumple $A \twoheadrightarrow D$.

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_1	c_2	d_1
a_2	b_2	c_2	d_2
a_2	b_2	c_3	d_3

Considerando las dos últimas tuplas de la instancia y aplicando la definición de *dependencias multivaluadas*, si se cumple $A \twoheadrightarrow D$ entonces también deberían existir en la instancia las tuplas $\langle a_2, b_2, c_2, d_3 \rangle$ y $\langle a_2, b_2, c_3, d_2 \rangle$. Como estas tuplas no se encuentran en la instancia podemos afirmar que la *dmv* $A \twoheadrightarrow D$ no se cumple en dicha instancia.

3. Usted fue contratado para crear una base de datos relacional con la información relativa a las consultas realizadas por los clientes de una veterinaria. El resumen del relevamiento obtenido sobre esa parte de la realidad es el siguiente:

De los veterinarios interesa la cédula (ciVet), el nombre (nomVet), el teléfono (telVet) y los horarios de los días de semana en los que realiza consultas (diaHoraVet). De las mascotas interesa la matrícula (matMas) y un número de teléfono del dueño (telDue). Respecto a la consulta entre un cliente y un veterinario es importante saber la fecha y la hora (fechaHoraCons) y, en caso de que haya pasado esa fecha, un comentario del veterinario (comVet) que realizó dicha consulta. Un cliente nunca consulta por su mascota, a un mismo veterinario, más de una vez al día.

Por ejemplo:

El veterinario con cédula v1 atiende (diaHoraVet) todos los lunes a las 14:00 hs, los miércoles 15:30 hs. y los viernes 18:00 hs. El veterinario v1 atendió a la mascota C123 (fechaHoraCons) el viernes 15/11 a las 18:00 hs, el comentario del veterinario fue "consulta normal".

- 3.1 Indicar si se cumplen las siguientes dependencias:

$ciVet \rightarrow nomVet, telVet$; $matMas \rightarrow telDue, nroFich$ y $ciVet \twoheadrightarrow diaHoraVet$.
Justifique su respuesta.

SOLUCIÓN:

Dependencia	Se cumple?
$ciVet \rightarrow nomVet, telVet$	Si, ya que la cédula identifica a cada veterinario y, por tanto, determina su nombre y teléfono.
$matMas \rightarrow telDue$	Si, ya que la matrícula identifica a las mascotas, y por tanto, determina el teléfono del dueño de la misma.
$ciVet \twoheadrightarrow diaHoraVet$	Si, ya que el hecho de que un veterinario atienda en varios horarios de la semana, no depende de la mascota a la que atiende.

- 3.2 ¿El conjunto F de la parte 3.1 incluye todas las dependencias funcionales y multivaluadas que se desprenden de la realidad descrita? En caso negativo agregue las dependencias faltantes a F justificando su respuesta. No considere dependencias que se derivan de F.

SOLUCIÓN:

No incluye la dependencia funcional que refleja el hecho de que un veterinario solo puede atender una mascota a la vez:

$ciVet, fechaHoraCons \rightarrow matMas$

- 3.3 Indicar cuál es la máxima forma normal en que se encuentra el esquema universal en función del F resultante de la parte 3.2.

SOLUCIÓN:

Resumiendo, se tiene el conjunto de dependencias:

$$F = \{ \text{ciVet} \rightarrow \text{nomVet}, \text{telVet}; \quad \text{matMas} \rightarrow \text{telDue}; \quad \text{ciVet} \twoheadrightarrow \text{diaHoraVet}; \\ \text{ciVet}, \text{fechaHoraCons} \rightarrow \text{matMas} \}$$

sobre el esquema $R = (\text{ciVet}, \text{nomVet}, \text{telVet}, \text{diaHoraVet}, \text{matMas}, \text{telDue}, \text{fechaHoraCons})$

Dado que ciVet no se encuentra a la derecha de ninguna dependencia, entonces pertenece a toda clave de R . Además, dado que nomVet , telVet , telDue , y diaHoraVet no figuran en el lado izquierdo de ninguna dependencia, entonces no pertenecen a ninguna clave de R .

Calculando la clausura de ciVet bajo F se obtiene:

$$\{\text{ciVet}\}_F^+ = \{\text{ciVet}, \text{nomVet}, \text{telVet}\}$$

Por tanto, ciVet no es clave de R . Luego, dado que nomVet no es primo y ciVet no es superclave de R , se tiene que la dependencia $\text{ciVet} \rightarrow \text{nomVet}$ viola 3NF. Más aún, esta dependencia viola 2NF, ya que un atributo no primo depende de forma parcial de alguna clave de R . En conclusión, el esquema universal R está en 1NF.

Ejercicio 2 (12 puntos)

Sea el esquema relación $R(A, B, C, D, E, G, H, I, J, K, L)$ y F el un conjunto de dependencias funcionales sobre R :

$$F = \{ A \rightarrow BCD, I \rightarrow AEJKL, ECK \rightarrow L \}$$

Considere además la siguiente descomposición ρ :

$$R_1(A, B, C, E, G, H, I, J, K, L)$$

$$R_2(C, I, D)$$

Se pide:

1. Calcular todas las claves de R . Justificar.

SOLUCION:

Por propiedades vistas en el curso:

- B, D, J, L no están en ninguna clave porque, en las df , sólo aparecen a la derecha.
- I, G, H están en toda clave porque, en las df , no aparecen del lado derecho.

Si calculamos $\{I, G, H\}^+$ vemos que $\{I, G, H\}$ determina a todos los atributos de R y por lo tanto es clave. Por propiedad sabemos que IGH debe estar en toda clave y si agregamos atributos al conjunto IGH obtendremos conjuntos de atributos que determinan a todos los atributos de R pero no son minimales, por lo tanto (IGH) es la única clave en R .

2. Determinar si ρ tiene JSP. Justifique su respuesta.

SOLUCION:

Para determinar si ρ tiene JSP o bien se aplica el teorema visto en el curso para descomposiciones en dos subesquemas o se aplica el método general (matriz).

Aplicaremos el teorema visto en el curso, el cual se enuncia a continuación:

Sea R un esquema relacional, F un conjunto de dependencias funcionales que se cumple en R y $\rho(R_1, R_2)$ una descomposición de R .

ρ cumple con la propiedad de JSP sii se verifica alguna de las siguientes dependencias funcionales en F^ : $(R_1 \cap R_2) \rightarrow (R_2 - R_1)$ o $(R_1 \cap R_2) \rightarrow (R_2 - R_1)$*

En este caso $R_1 \cap R_2 = \{C, I\}$ y $R_2 - R_1 = \{D\}$

Para probar si $CI \rightarrow D$ se cumple en F^+ o calculamos la clausura de CI y vemos si D pertenece a dicho conjunto o intentamos derivar la dependencia $CI \rightarrow D$ a partir de las dfs de F y las reglas de inferencia vistas en el curso.

Calculando $(CI)^+_F$ vemos que D pertenece a este conjunto por lo tanto $CI \rightarrow D$ se cumple en F^+ .

También puede demostrarse sin calcular la clausura. Como $I \rightarrow A$ y $A \rightarrow D$ se cumplen en F , por transitiva $I \rightarrow D$ se cumple en F^+ y luego por aumentación $CI \rightarrow CD$ y por descomposición $CI \rightarrow D$ se cumple en F^+

3. Proyectar dependencias funcionales de R sobre ρ indicando si se pierden dependencias. Justifique su respuesta.

SOLUCION:

$\Pi_{R_1}(F) = \{A \rightarrow BC; I \rightarrow AEJKL; ECK \rightarrow L\}$

$\Pi_{R_2}(F) = \{I \rightarrow CD\}$

Esta dependencia surge porque $I \rightarrow A$ y $A \rightarrow CD$, por transitividad.

Podemos probar que $A \rightarrow D$ se pierde ya que D no pertenece a $A^+_{\Pi_{R_1}(F) \cup \Pi_{R_2}(F)}$.

4. Aplique el algoritmo para encontrar una descomposición en BCNF sobre R_1 .

SOLUCION:

Calculando $A^+_{\Pi_{R_1}(F)} = \{A, B, C\}$ vemos que A no es superclave en R_1 ya que su clausura no contiene a todos los atributos del esquema. Por lo tanto $A \rightarrow BC$ viola BCNF ya que no incluye una superclave del lado izquierdo. Partimos R_1 por esa dependencia obteniendo: $R_{11}(A, B, C)$ y $R_{12}(A, E, G, H, I, J, K, L)$.

$\Pi_{R_{11}}(F) = \{A \rightarrow BC\}$

Como $A^+_{\Pi_{R_{11}}(F)} = \{A, B, C\} \Rightarrow A$ determina a todos los atributos de R_{11} y por lo tanto es clave en $R_{11} \Rightarrow A$ es superclave en $R_{11} \Rightarrow$ **R_{11} está en BCNF**

$\Pi_{R_{12}}(F) = \{I \rightarrow AEJKL, AEK \rightarrow L\}$

Como $I^+_{\Pi_{R_{12}}(F)} = \{I, A, E, J, K, L\}$ no contiene a todos los atributos de R_{12} (faltan G y H), I no es clave en $R_{12} \Rightarrow I$ no es superclave en $R_{12} \Rightarrow I \rightarrow AEJKL$ viola BCNF en R_{12}

Partimos $R_{12}(A, E, G, H, I, J, K, L)$ por $I \rightarrow AEJKL$ y obtenemos

$R_{121}(G, H, I)$ y $R_{122}(A, E, I, J, K, L)$

$\Pi_{R_{121}}(F) = \{\} \Rightarrow$ **R_{121} está en BCNF**

$\Pi_{R_{122}}(F) = \{I \rightarrow AEJKL, AEK \rightarrow L\}$

Como $AEK^+_{\Pi_{R_{122}}(F)} = \{A, E, K, L\}$ no contiene a todos los atributos de R_{122} , AEK no es clave en $R_{122} \Rightarrow AEK$ no es superclave en $R_{122} \Rightarrow AEK \rightarrow L$ viola BCNF en R_{122}

Partimos $R_{122}(A, E, I, J, K, L)$ por $AEK \rightarrow L$ y obtenemos

$R_{1221}(A, E, I, J, K)$ y $R_{1222}(A, E, K, L)$

$\Pi_{R_{1221}}(F) = \{I \rightarrow AEJK\}$

$\Pi_{R_{1222}}(F) = \{AEK \rightarrow L\}$

Como $I^+_{\Pi_{R_{1221}}(F)} = \{I, A, E, J, K\} \Rightarrow$ **R_{1221} está en BCNF**

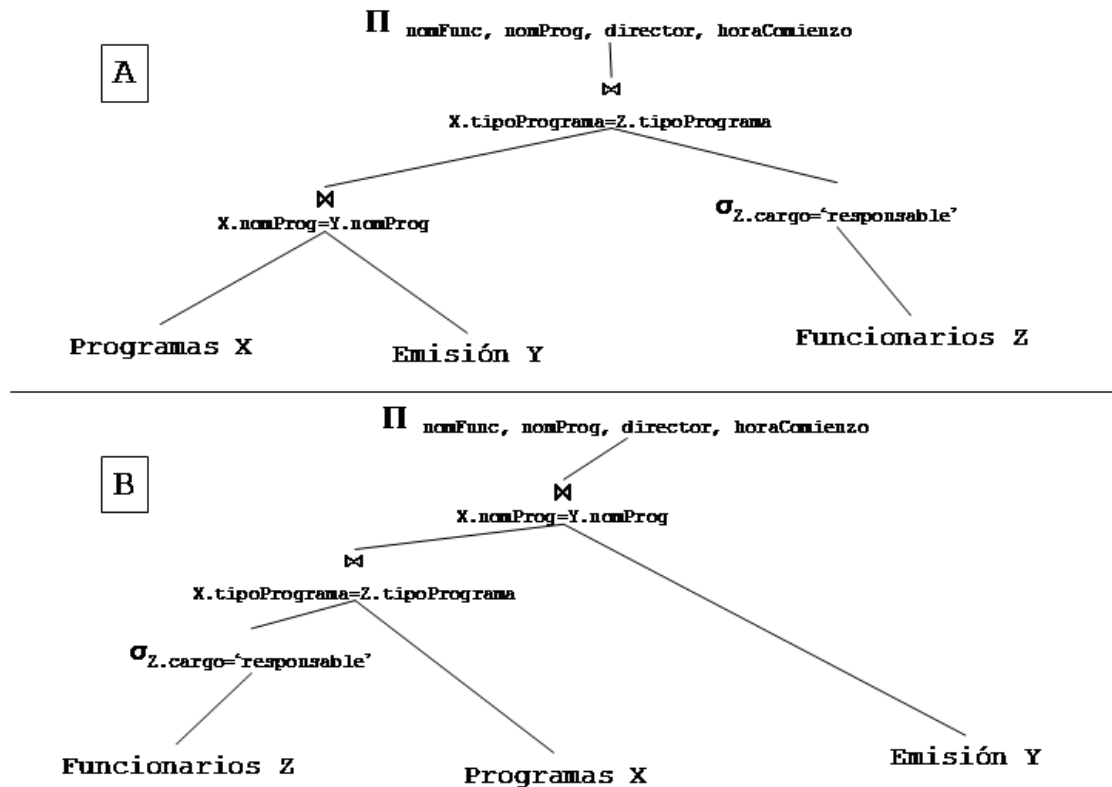
Como $AEK^+_{\Pi_{R_{1222}}(F)} = \{A, E, K, L\} \Rightarrow$ **R_{1222} está en BCNF**

Por todo lo anterior la siguiente es una descomposición de R_1 en BCNF:

$\rho = \{R_{11}(A, B, C), R_{121}(G, H, I), R_{1221}(A, E, I, J, K), R_{1222}(A, E, K, L)\}$

Ejercicio 3 (18 puntos)

Un optimizador generó los siguientes planes lógicos en diferentes momentos para la misma consulta SQL:



Las tablas de la base son las siguientes:

PROGRAMAS (nomProg, tipoPrograma, director)
 EMISIÓN (nomProg, dia, horaComienzo, horaFin)
 FUNCIONARIOS (ciFunc, nomFunc, dirFunc, telFunc, tipoPrograma, cargo, horaEntrada, horaSalida)

1. Escriba la consulta SQL de la cual surgió el plan. Justifique su respuesta.

SOLUCIÓN

Para construir la consulta SQL primero analizaremos cómo las diferentes partes de la consulta (o cláusulas) dan origen al árbol canónico, del cual luego se obtienen planes lógicos aplicando un conjunto de heurísticas.

La consulta SQL es una consulta de la forma SELECT... FROM .. WHERE donde:

- a) en la cláusula SELECT aparece un conjunto de atributos, los cuales aparecen en el árbol canónico como atributos en una operación de proyección la cual se coloca en la raíz del árbol.
- b) en la cláusula FROM aparece un conjunto de tablas, las cuales son las hojas del árbol canónico y aparecen en el árbol en el mismo orden en que aparecen en la cláusula FROM

- c) en la cláusula WHERE aparece una condición booleana, la cual suele ser una conjunción de condiciones. Esta condición da origen a una operación de selección en el árbol canónico.

En ambos planes lógicos vemos que hay una sola operación de proyección (en la raíz de cada árbol) así que podemos asumir que los atributos que aparecen en la proyección son los que aparecían en la cláusula SELECT de la consulta original.

En cada plan lógico hay un par de operaciones de join. Sabemos que estas operaciones surgen al aplicar las heurísticas de combinar operaciones de selección con productos cartesianos. De esto podemos concluir que las condiciones que aparecen en los joins eran parte de la conjunción de condiciones que aparecía en la cláusula WHERE de la consulta.

También observamos que en cada plan lógico hay una operación de selección (cargo = "Responsable"). Esta operación (así como las que se usaron para combinar con los productos cartesianos) son el resultado de transformar la selección original, con una condición expresada como la conjunción de condiciones atómicas, en una cascada de selecciones atómicas. De esto podemos concluir que esta condición también estaba en el WHERE de la consulta original.

Resta saber que había en la cláusula FROM. Para esto observamos las hojas de los planes y vemos que hay 3 tablas. Lo que no podemos asegurar es el orden en que aparecían en la consulta, ya que las hojas de los planes lógicos se reordenan para satisfacer criterios de tamaño.

Por todo lo anterior la consulta SQL a continuación podría ser la consulta para la cual se obtuvieron los planes lógicos presentados

```
SELECT nomFunc, nomProg, director, horaComienzo
FROM Programas X, Emision Y, Funcionarios Z
WHERE X.tipoPrograma=Z.tipoPrograma and
      X.nomProg=Y.nomProg and
      Z.cargo='responsable'
```

2. Considerando los siguientes datos del catálogo, decida cual de esos planes es mejor. Justifique su respuesta.

	PROGRAMAS	EMISION	FUNCIONARIOS
Cantidad tuplas	300	800	100
Distribución de valores	Existen 50 tipoPrograma diferentes (periodístico, telenovela, documental, etc.).	Existen 300 programas diferentes.	Existe un responsable de emisión diferente por cada tipoPrograma y todos los tipos de Programas tienen un responsable de emisión.

Nota: Cada clave primaria tiene un índice primario.

A continuación se calculan los tamaños para cada uno de los planes lógicos. Podemos observar que en el plan B el tamaño de la rama izquierda de cada operación de join es menor que el de la rama derecha, por lo tanto es un mejor plan (según los criterios vistos en el curso) que el plan A dado que en éste no se cumple este criterio.

SOLUCIÓN:

Parte 1.

Para responder a estas afirmaciones se toman en cuenta las definiciones de historia estricta, historia que evita abortos en cascada, historia recuperable e historia serializable, los cuales pueden consultarse en el material teórico del curso.

a) **No es posible.** Que una historia sea estricta implica que evita abortos en cascada.

b) **Sí es posible.** Un ejemplo puede ser el siguiente:

$T_1: w_1(X)c_1$

$T_2: r_2(X)c_2$

H: $w_1(X)r_2(X)c_2c_1$

T_1 lee de T_2 , pero T_2 confirma antes que T_1 . Esto hace que la historia no cumpla con la definición de recuperable, por ende que no evite abortos en cascada y que no sea estricta.

La historia es serializable porque su grafo de seriabilidad es acíclico (sólo tiene un arco desde T_1 a T_2).

c) **Sí es posible.** Un ejemplo puede ser el siguiente:

$T_1: w_1(X)w_1(Y)c_1$

$T_2: r_2(Y)r_2(X)c_2$

H: $w_1(X)r_2(Y)w_1(Y)c_1r_2(X)c_2$

La historia es estricta porque ninguna transacción lee o escribe hasta que todas las transacciones que escribieron ese ítem fueron confirmadas (caso de X). Como es estricta, entonces es recuperable.

La historia es no serializable porque su grafo de seriabilidad tiene ciclos (un arco va de T_1 a T_2 y el otro va de T_2 a T_1).

d) **No es posible.** Que una historia sea estricta implica que sea recuperable.

Parte 2.1

En H_1 se identifican las siguientes transacciones:

$T_1: w_1(X)w_1(X)u_1(X)r_1(Y)r_1(Y) u_1(Y) c_1$

$T_2: r_2(X)r_2(X)u_2(X)r_2(Y)r_2(Y)u_2(Y)w_2(Z)w_2(Z)u_2(Z)c_2$

$T_3: r_3(Z)r_3(Z)u_3(Z)w_3(Y)w_3(Y)u_3(Y)c_3$

En H_2 se identifican las siguientes transacciones:

$T_1: w_1(X)w_1(X)r_1(Y)r_1(Y)u_1(X) u_1(Y)c_1$

$T_2: r_2(X)r_2(X)r_2(Y)r_2(Y)w_2(Z)w_2(Z)u_2(Z)u_2(X)u_2(Y)c_2$

$T_3: r_3(Z)r_3(Z) w_3(Y)w_3(Y)u_3(Y)u_3(Z)c_3$

Parte 2.2

A continuación se analizan las transacciones de H_1 :

T_1 : No sigue 2PL. No cumple con las fases de crecimiento y contracción. Ver $u_1(X)r_1(Y)$.

T_2 : No sigue 2PL. No cumple con las fases de crecimiento y contracción. Ver $u_2(X)r_2(Y)$.

T_3 : No sigue 2PL. No cumple con las fases de crecimiento y contracción. Ver $u_3(Z)w_3(Y)$.

A continuación se analizan las transacciones de H_2 :

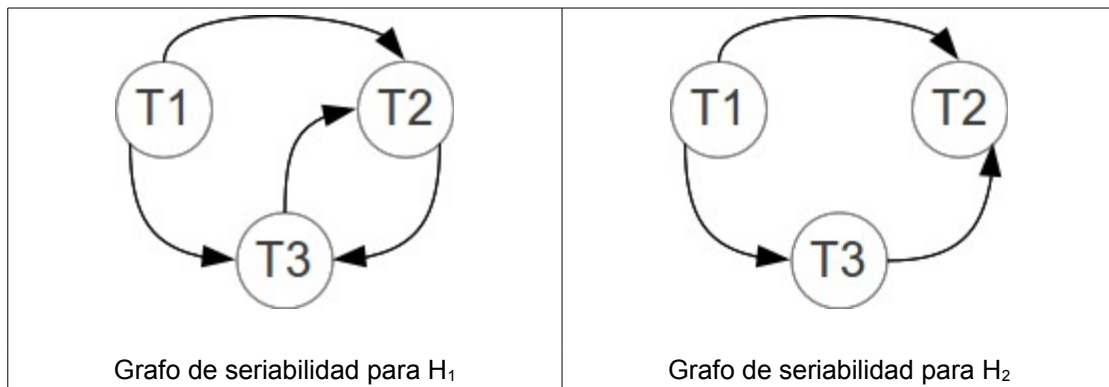
T_1 : Sigue 2PL riguroso, ya que no libera ningún read lock y write lock hasta que termina la transacción (notar que luego de $u_1(X)u_1(Y)$ sólo confirma la transacción).

T_2 : Sigue 2PL riguroso, ya que no libera ningún read lock y write lock hasta que termina la transacción (notar que luego de $u_2(Z)u_2(X)u_2(Y)$ sólo confirma la transacción).

T_3 : Sigue 2PL riguroso, ya que no libera ningún read lock y write lock hasta que termina la transacción (notar que luego de $u_3(Y)u_3(Z)$ sólo confirma la transacción).

Parte 2.3

Para verificar si H_1 y H_2 son serializables se elabora el grafo de seriabilidad para cada historia.



Como el grafo de seriabilidad de H_1 tiene un ciclo, entonces H_1 no es serializable.

Como el grafo de seriabilidad de H_2 es acíclico, entonces H_2 es serializable.

Parte 2.4

La única historia serializable es H_2 , pero como todas sus transacciones siguen 2PL y este protocolo asegura seriabilidad, no es posible construir una historia no serializable con las transacciones de H_2 .