

Práctica 1 - Diseño de modelo relacional

Realizado por:

- Francisco Rodríguez Cuenca
- Alejadro Lozano Morales

Ejercicio 1

Determinar el grado de normalización de estas dos relaciones, indicando las claves existentes. Se asume que el conjunto de dependencias L1 y L2 representan, de manera independiente, un recubrimiento mínimo y no redundante.

- Primera tabla
 - Clasificación de atributos

Esenciales = {H,L}
Posibles = {C,A}
No posibles = {B,D,E,F,G,I,J,K,M,N,O,P,Q,R}

- Claves finales

Clave = HL

- Forma Normal

Primera forma normal ya que $H \rightarrow I$, siendo H subclave

- Segunda tabla

- Clasificación de atributos

Esenciales = {A,S}
Posibles = {C}
No posibles = {B,D,E,F,G,T,U,V,W}

- Claves finales

Clave = AS

- Forma Normal

Primera forma normal ya que A → C, siendo A subclave

Ejercicio 2

Modificar con SQL el esquema de las dos tablas EXCLUSIVAMENTE para dar de alta la clave primaria elegida para cada una de ellas en el apartado anterior

```
ALTER TABLE primeratabla ADD CONSTRAINT pk_primeratabla PRIMARY
KEY(player_id,play_id);

ALTER TABLE segundatabla ADD CONSTRAINT pk_segundtabla PRIMARY
KEY(game_id,team_id);
```

Ejercicio 3

Resolver en SQL las siguientes consultas:

a. ¿Cuántos partidos ha jugado en el año 2013 el equipo de nombre Blackhawks?

```
USE practica1bda;
SELECT COUNT(*) AS Num_Partidos_Jugados
FROM segundatabla s
WHERE year(date_time)='2013'
AND teamName= 'BlackHawks'
```

b. Por cada partido registrado en el mes de enero de 2013, mostrar el nombre de los equipos locales

```
USE practica1bda;
SELECT date_time AS Fecha_Partido, s.TeamName AS EquipoLocal
FROM segundatabla s
WHERE year(date_time)='2013'
AND month(date_time)='01'
AND s.team_id=s.home_team_id;
```

c. Por cada equipo, mostrar su identificador, nombre, y estadísticas (tgoals, tshots y thits) en los partidos jugados en 2013 como equipos locales

```
USE practica1bda;
SELECT s.TeamName AS EquipoLocal,s.tgoals AS Goles,s.tshots AS Tiros,s.thits
AS Golpes,s.date_time AS Fecha_Partido
```

```

FROM segundatable s
WHERE year(date_time)='2013'
AND s.team_id=s.home_team_id;

```

d. Obtener, para cada partido que haya jugado el o los equipo(s) en los que haya jugado el jugador Adam McQuaid, el identificador del equipo, el nombre del equipo, el año, mes y día del partido y el identificador de cada jugada realizada junto con su descripción (event) para cada una de las jugadas que haya realizado el equipo en cada partido (event) ordenadas por la fecha del partido

```

USE practica1bda;
SELECT p.date_time,p.play_id,p.event, s1.teamName AS NombreAgainst,
s2.teamName AS NombreFor
FROM primeratabla p
INNER JOIN segundatable s1 ON s1.team_id = p.team_id_against
INNER JOIN segundatable s2 ON s2.team_id = p.team_id_for
WHERE p.firstName='Adam'
AND p.lastname = 'McQuaid'
GROUP BY p.team_id_for,p.team_id_against, p.event
ORDER BY p.date_time;

```

Hemos realizado dos copias de segundatable (s1 y s2) para poder obtener simultáneamente los nombres del equipo_for y el equipo_against, ya que con una sola copia no podríamos realizarlo.

Ejercicio 4

Insertar mediante SQL, si es posible, (manteniendo las claves primarias creadas) los siguientes datos, explicando los problemas y anomalías encontradas en caso de haberlos:

a. Los datos del nuevo jugador que se llama Samuel Smithson y es canadiense (CAN) y al que se le ha asignado el identificador 5554441

```

USE practica1bda;

INSERT INTO primeratabla (firstname,lastname,nationality,player_id)
VALUES ('Samuel','Smithson','CAN',5554441);

```

Al ser una clave compuesta por Player_id(H) y Play_id(L) y tener solo valor para player_id(H) nos da una ANOMALIA DE INSERCIÓN por falta de valor en clave principal(HL)

b. El partido de código 2012030999 entre los “Red Wings” (locales) y los “Penguins” que terminó 3 a 2 a favor de los locales

```
USE practica1bda;

INSERT INTO segundatabla
(game_id,team_id,home_goals,away_goals,teamName,home_team_id,away_team_id)
VALUES (2012030999,
        (SELECT s.team_id
         FROM segundatabla s
         WHERE s.teamName = 'Red Wings'
         GROUP BY s.team_id)
      ,3,2,'Red Wings',
        (SELECT s.team_id
         FROM segundatabla s
         WHERE s.teamName = 'Red Wings'
         GROUP BY s.team_id)
      ,(SELECT s.team_id
         FROM segundatabla s
         WHERE s.teamName = 'Penguins'
         GROUP BY s.team_id));

INSERT INTO segundatabla
(game_id,team_id,home_goals,away_goals,teamName,home_team_id,away_team_id)
VALUES (2012030999,
        (SELECT s.team_id
         FROM segundatabla s
         WHERE s.teamName = 'Penguins'
         GROUP BY s.team_id)
      ,3,2,'Red Wings',
        (SELECT s.team_id
         FROM segundatabla s
         WHERE s.teamName = 'Red Wings'
         GROUP BY s.team_id)
      ,(SELECT s.team_id
         FROM segundatabla s
         WHERE s.teamName = 'Penguins'
         GROUP BY s.team_id));
```

En este caso al tener valores para la clave compuesta no nos da ningún error al insertar, sin embargo, los datos identificativos los hemos tenido que obtener a partir de consultas.

c. Los datos de un nuevo equipo, denominado “Lasters” y cuyo código será 99

```
USE practica1bda;
```

```
INSERT INTO segundatable(teamName,team_id)
VALUES ('Lasters',99);
```

ANOMALIA DE INSERCIÓN por falta de valor en game_id(A) y al ser AS clave principal compuesta

Ejercicio 5

Actualizar el resultado del partido de código '2012020053' ya que el partido terminó realmente 5 a 3 en vez de 6 a 4 como figura. Comentar problemas y anomalías si las hay.

```
USE practica1bda;

UPDATE segundatable SET home_goals=5,away_goals=3
WHERE game_id=2012020053;
```

Ejercicio 6

A partir de febrero de 2013, el equipo denominado "Wild" pasa a denominarse "Wild Side". Reflejar este hecho en las tablas, explicando problemas y anomalías. Comentar si surge algún problema entre datos en las tablas y las dependencias funcionales.

```
USE practica1bda;

UPDATE segundatable SET teamName='Wild Side'
WHERE team_id in (SELECT s.team_id
                   FROM segundatable s
                   WHERE s.teamname = 'Wild'
                   GROUP BY s.team_id)
      AND month(date_time)>= '02'
      AND year(date_time)>= '2013'
      AND teamName='Wild';
```

Ejercicio 7

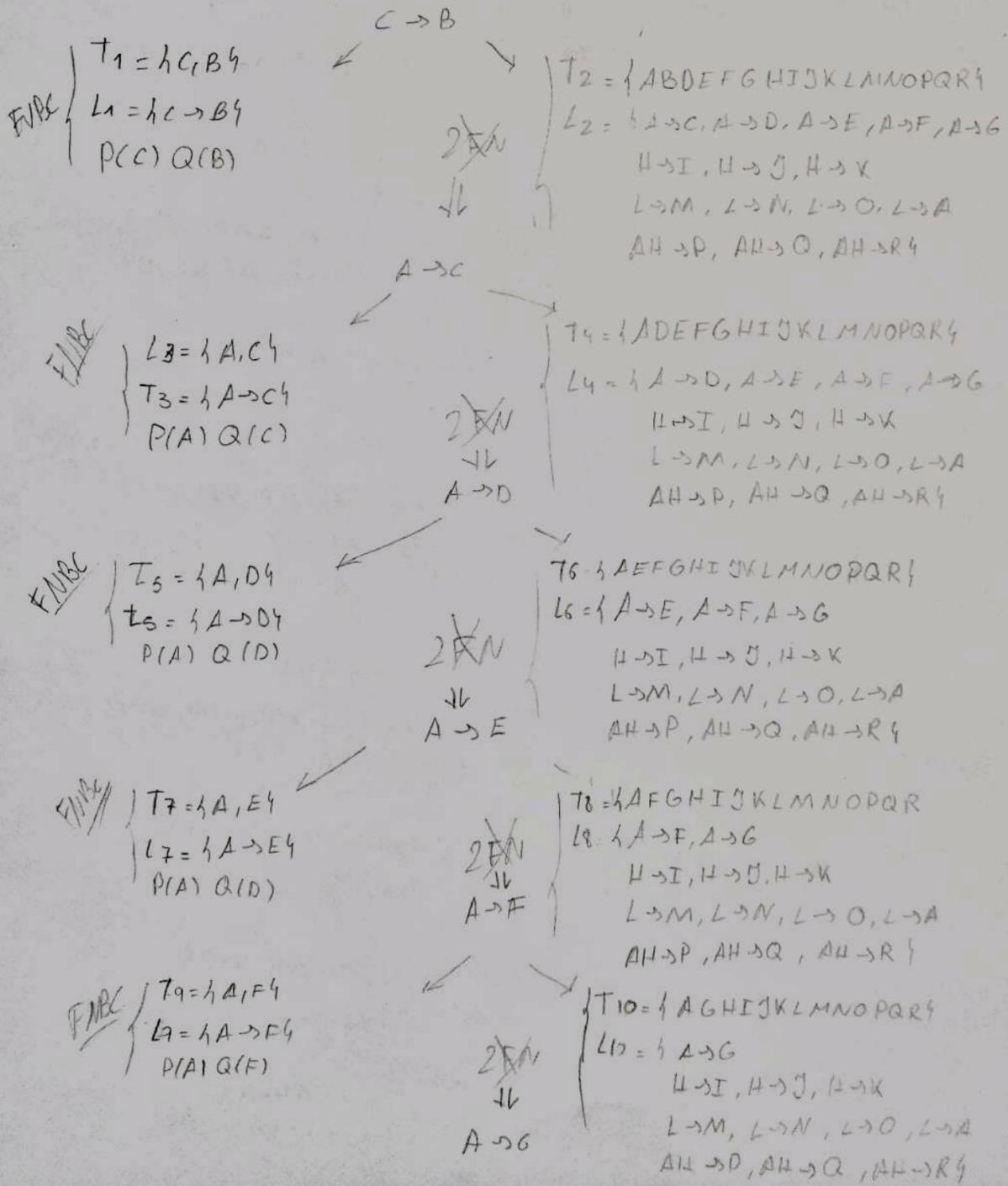
Normalizar R hasta conseguir un conjunto de relaciones que estén en FNBC mediante el algoritmo de descomposición con LJ. Explicar todos los pasos dados.

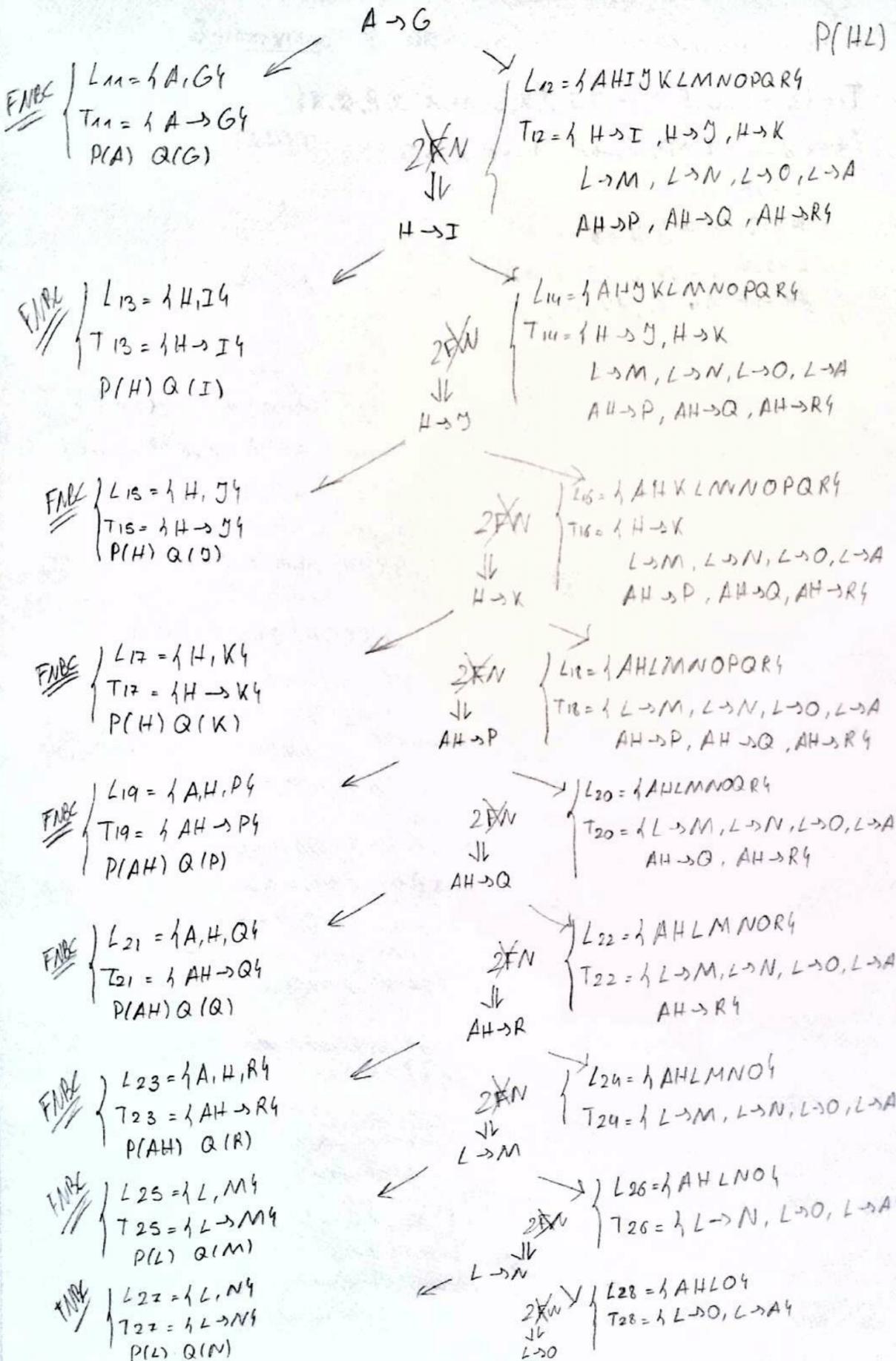
→ Proceso normalización por omisiones de primera tabla

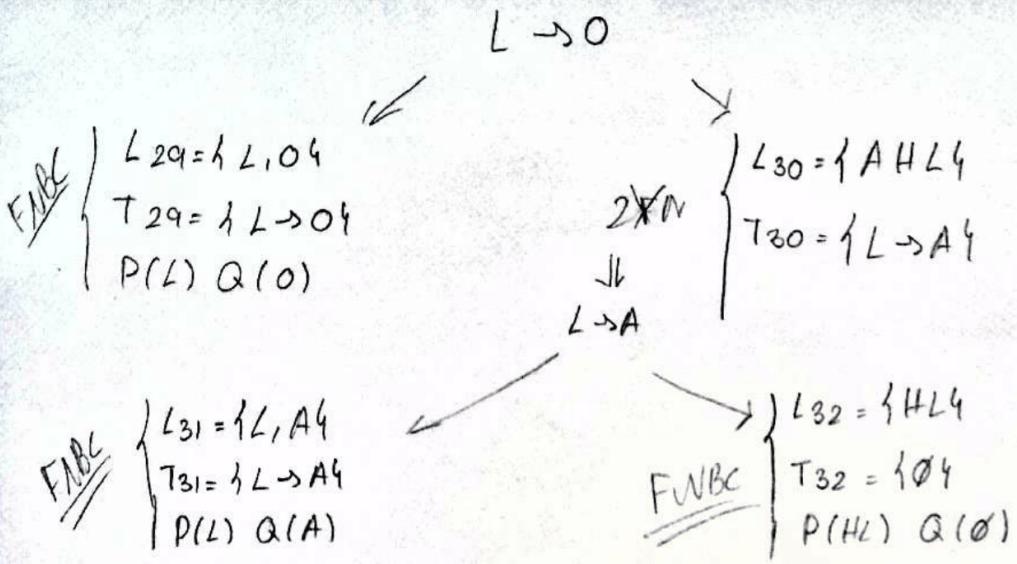
$$T_1 = \{A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R\}$$

$$L_1 = \{A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F, A \rightarrow G, C \rightarrow B, P(HL)$$

$$\begin{aligned} H &\rightarrow I, H \rightarrow J, H \rightarrow K, \\ L &\rightarrow M, L \rightarrow N, L \rightarrow O, L \rightarrow A \\ AH &\rightarrow P, AH \rightarrow Q, AH \rightarrow R \end{aligned}$$





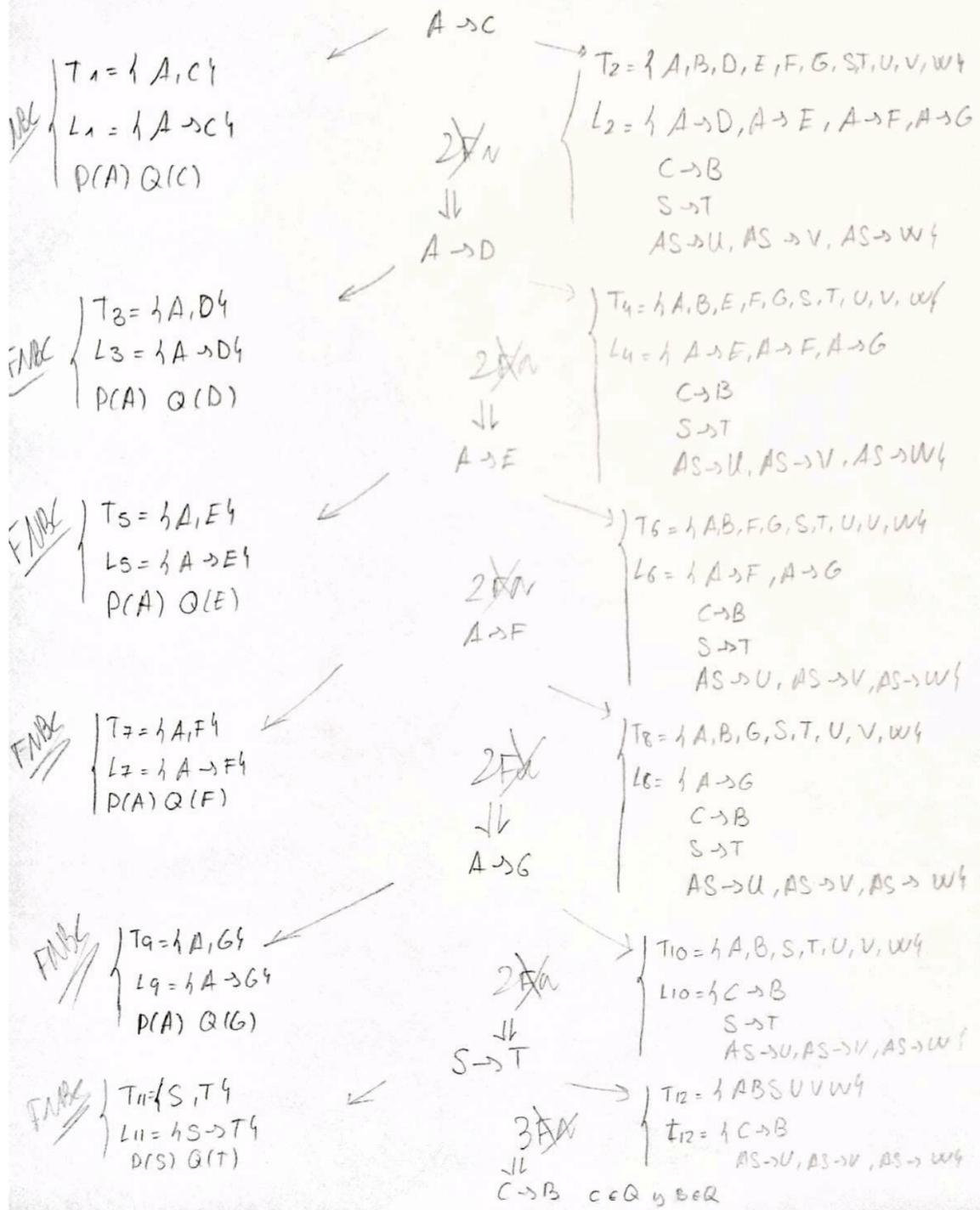


→ Proceso normalización por análisis de segundatanda

$$T = \{A, B, C, D, E, F, G, S, T, U, V, W\} \quad P(AS)$$

$$L = \{A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F, A \rightarrow G, \\ C \rightarrow B, \\ S \rightarrow T, \\ AS \rightarrow U, AS \rightarrow V, AS \rightarrow W\}$$

~~2AN A → C y C ∈ Q~~



$$\begin{array}{c}
 C \rightarrow B \\
 \swarrow \quad \searrow \\
 \begin{array}{l}
 \text{FNR} \left\{ \begin{array}{l} T_{13} = \{C, B\} \\ L_{13} = \{C \rightarrow B\} \\ P(C) Q(B) \end{array} \right. \\
 \hline
 \text{FNBC} \left\{ \begin{array}{l} T_{14} = \{ASUVW\} \\ L_{14} = \{AS \rightarrow U, AS \rightarrow V, AS \rightarrow W\} \\ P(AS) Q(U, V, W) \end{array} \right.
 \end{array}
 \end{array}$$

Las soluciones se pueden agrupar en las siguientes tablas:

T	L
H,I,J,K	H→I, H→J, H→K
S,T	S→T
B,C	C→B
A,H,P,Q,R	AH→P, AH→Q, AH→R
A,S,U,V,W	AS→U, AS→V, AS→W
A,L,M,N,O	L→M, L→N, L→O, L→A
A,C,D,E,F,G	A→C, A→D, A→E, A→F, A→G
H,L	

Ejercicio 8

Normalizar R mediante el algoritmo de síntesis para conseguir un conjunto de relaciones que estén, al menos, en 3FN. Explicar los pasos dados y comparar el resultado con el punto 7.

Primera Tabla

T	L
A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R	A→C,A→D,A→E,A→F,A→G,C→B,H→I,H→J,H→K,L→M,L→N,L→O,L→A, AH→P,AH→Q,AH→R

Recubrimiento mínimo

El recubrimiento mínimo ya está calculado y nos lo da el enunciado

Cálculo claves

La clave de la primera tabla es HL

Cálculo de la forma normal

Está en primera forma normal

Agrupar dependencias que tengan el mismo implicante

T	L
H,I,J,K	$H \rightarrow I, H \rightarrow J, H \rightarrow K$
B,C	$C \rightarrow B$
A,H,P,Q,R	$AH \rightarrow P, AH \rightarrow Q, AH \rightarrow R$
A,L,M,N,O	$L \rightarrow M, L \rightarrow N, L \rightarrow O, L \rightarrow A$
A,C,D,E,F,G	$A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F, A \rightarrow G$
S,T	$S \rightarrow T$

Segunda Tabla

T	L
A,B,C,D,E,F,G,S,T,U,V,W	$A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F, A \rightarrow G, C \rightarrow B, S \rightarrow T, AS \rightarrow U, AS \rightarrow V, AS \rightarrow W$
A,S,T,U,V,W	$AS \rightarrow U, AS \rightarrow V, AS \rightarrow W$

Recubrimiento mínimo

El recubrimiento mínimo ya está calculado y nos lo da el enunciado

Cálculo claves

La clave de la primera tabla es AS

Cálculo de la forma normal

Está en primera forma normal

Agrupar dependencias que tengan el mismo implicante

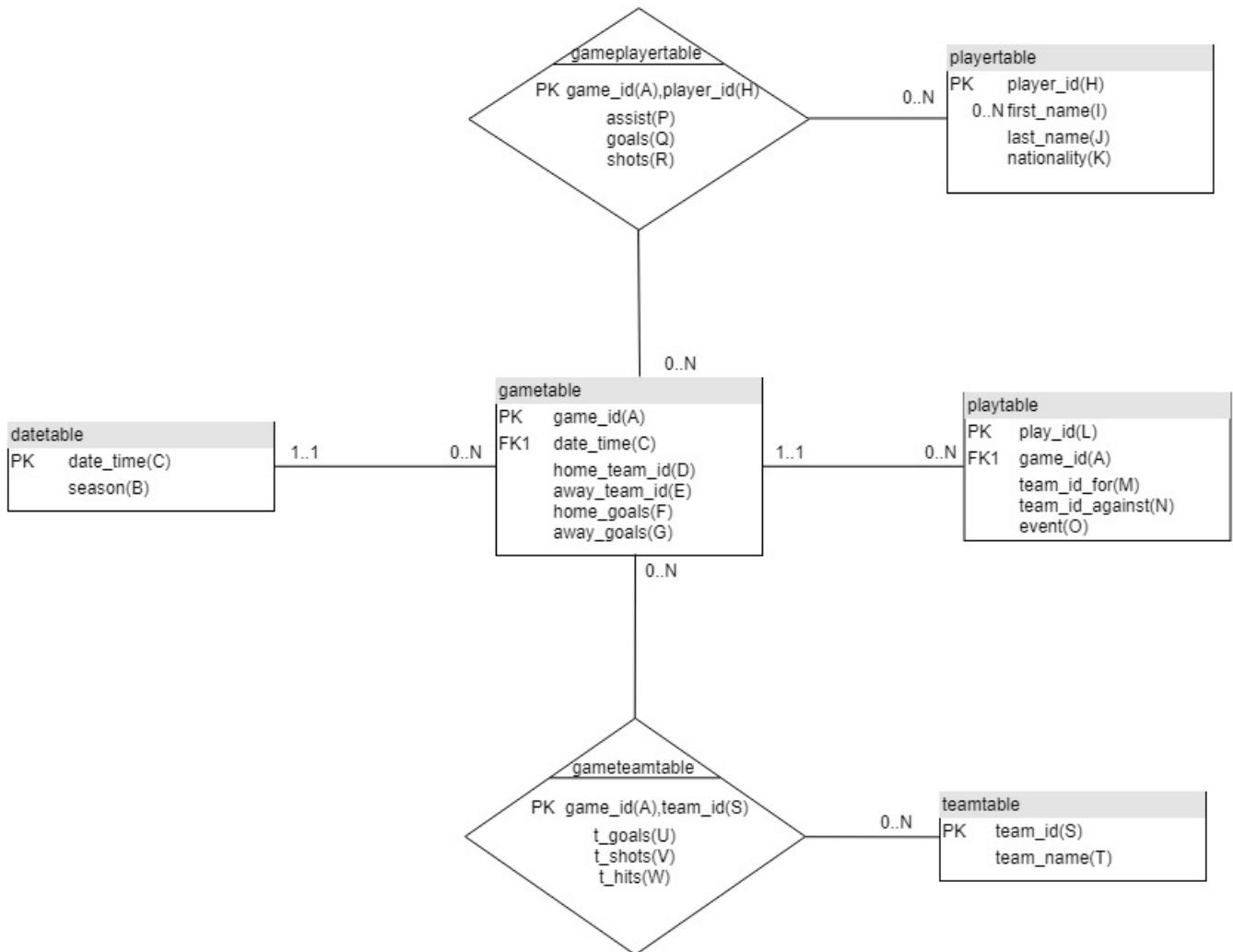
T	L
B,C	$C \rightarrow B$
A,C,D,E,F,G	$A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F, A \rightarrow G$
A,S,U,V,W	$AS \rightarrow U, AS \rightarrow V, AS \rightarrow W$

Al final agrupando las tablas creadas sale el siguiente resultado:

T	L
B,C	C→B
A,C,D,E,F,G	A→C, A→D, A→E, A→F, A→G
A,S,U,V,W	AS→U, AS→V, AS→W
H,I,J,K	H→I, H→J, H→K
A,H,P,Q,R	AH→P, AH→Q, AH→R
A,L,M,N,O	L→M, L→N, L→O, L→A
S,T	S→T

Ejercicio 9

Dibujar un modelo entidad/relación que se corresponda con lo obtenido en el paso 7 y, si se encuentra alguna irregularidad, revisar el diseño obtenido realizando justificadamente los cambios que se estimen oportunos sin perder la normalización efectuada.



Ejercicio 10

Escribir un script SQL para crear las tablas resultantes ya normalizadas y revisadas del punto 9, con sus correspondientes claves primarias y foráneas en la base de datos “practica1bda”.

```
CREATE DATABASE IF NOT EXISTS `practica1bda`;
USE `practica1bda`;

DROP TABLE IF EXISTS `playplayertable`;
DROP TABLE IF EXISTS `playtable`;
DROP TABLE IF EXISTS `playergametable`;
DROP TABLE IF EXISTS `teamgametable`;
DROP TABLE IF EXISTS `gametable`;
DROP TABLE IF EXISTS `teamtable`;
DROP TABLE IF EXISTS `timetable`;
DROP TABLE IF EXISTS `playertable`;

/*Previa creación de tablas incial*/
CREATE TABLE `playertable` (
  `player_id` int,
  `firstName` varchar(25),
  `lastName` varchar(25),
  `nationality` varchar(25),
  CONSTRAINT pk_playertable PRIMARY KEY(player_id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `timetable` (
  `season` int,
  `date_time` DATETIME,
  CONSTRAINT pk_timetable PRIMARY KEY(date_time)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `teamtable` (
  `team_id` int,
  `teamName` varchar(25),
  CONSTRAINT pk_teamtable PRIMARY KEY(team_id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `gametable` (
  `game_id` int,
  `date_time` DATETIME,
  `home_team_id` int,
  `away_team_id` int,
  `home_goals` int,
  `away_goals` int,
  CONSTRAINT pk_gametable PRIMARY KEY(game_id),
  CONSTRAINT fk_date_time_gametable FOREIGN KEY (date_time) REFERENCES
timetable(date_time)
```

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `teamgametable` (
  `game_id` int,
  `team_id` int,
  `tgoals` int,
  `tshots` int,
  `thits` int,
  CONSTRAINT pk_teamgametable PRIMARY KEY(game_id,team_id),
  CONSTRAINT fk_game_id_teamgametable FOREIGN KEY (game_id) REFERENCES
gametable(game_id),
  CONSTRAINT fk_team_id_teamgametable FOREIGN KEY (team_id) REFERENCES
teamtable(team_id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `playergametable` (
  `game_id` int,
  `player_id` int,
  `assists` int,
  `goals` int,
  `shots` int,
  CONSTRAINT pk_playergametable PRIMARY KEY(game_id,player_id),
  CONSTRAINT fk_game_id_playergametable FOREIGN KEY (game_id) REFERENCES
gametable(game_id),
  CONSTRAINT fk_player_id_playergametable FOREIGN KEY (player_id) REFERENCES
playertable(player_id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `playtable` (
  `game_id` int,
  `play_id` varchar(25),
  `team_id_for` int,
  `team_id_against` int,
  `event` varchar(25),
  CONSTRAINT pk_playtable PRIMARY KEY(play_id),
  CONSTRAINT fk_game_id_playtable FOREIGN KEY (game_id) REFERENCES
gametable(game_id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Ejercicio 11

Insertar los datos en las tablas normalizadas desde las tablas originales (ignorar los cambios de los apartados 4, 5 y 6, volviendo a crear las dos tablas desde los ficheros originales si es necesario) en sus correspondientes tablas mediante las correspondientes sentencias “`INSERT INTO ... SELECT ...`”.

```

INSERT INTO playertable(player_id,firstName,lastName,nationality)
(SELECT player_id,firstName,lastName,nationality
 FROM primeratabla
 GROUP BY player_id,firstName,lastName,nationality);

INSERT INTO timetable (date_time, season)
 (SELECT DISTINCT p.date_time,p.season
  FROM primeratabla p)
 UNION
 (SELECT DISTINCT s.date_time,s.season
  FROM segundatabla s);

INSERT INTO teamtable (team_id,teamName)
 (SELECT team_id,teamName
  FROM segundatabla
 GROUP BY team_id,teamName);

INSERT INTO
gametable(game_id,date_time,home_team_id,away_team_id,home_goals,away_goals)
 (SELECT
s.game_id,s.date_time,s.home_team_id,s.away_team_id,s.home_goals,s.away_goals
     FROM segundatabla s
 GROUP BY
s.game_id,s.date_time,s.home_team_id,s.away_team_id,s.home_goals,s.away_goals)
 UNION DISTINCT
 (SELECT
p.game_id,p.date_time,p.home_team_id,p.away_team_id,p.home_goals,p.away_goals
     FROM primeratabla p
 GROUP BY
p.game_id,p.date_time,p.home_team_id,p.away_team_id,p.home_goals,p.away_goals);

INSERT INTO teamgametable(game_id,team_id,tgoals,tshots,thits)
 (SELECT game_id,team_id,tgoals,tshots,thits
  FROM segundatabla
 GROUP BY game_id,team_id,tgoals,tshots,thits);

INSERT INTO playergametable(game_id,player_id,assists,goals,shots)
 (SELECT game_id,player_id,assists,goals,shots
  FROM primeratabla
 GROUP BY game_id,player_id,assists,goals,shots);

INSERT INTO playtable(game_id,play_id,team_id_for,team_id_against,event)
 (SELECT game_id,play_id,team_id_for,team_id_against,event
  FROM primeratabla
 GROUP BY game_id,play_id,team_id_for,team_id_against,event);

```

Ejercicio 12

Repetir los puntos 3 al 6 con la nueva base de datos. Comentar y justificar razonadamente las diferencias encontradas en cada punto.