

Práctica 1

Data Mart de Fútbol



Inteligencia de Negocio y Gestión de Procesos

Grado en Ingeniería Informática - Universitat d'Alacant / Universidad de Alicante
2020-21

Iván Mañús Murcia – 48729799K

Índice de contenido

1. Especificación de datos	4
2. Esquema Conceptual	5
2.1 Diferencias entre esquema estrella y copos de nieve	5
3. Esquema Lógico	6
4. Esquema Físico	7
5. Pentaho Schema Workbench	7
6. PDI (Pentaho Data Integration)	10
6.1 Entrada	10
6.2 Salida	11
6.3 Tabla de hechos	13
7. Pentaho Server	15
8. PowerBI	18
9. Bibliografía	23

Índice de ilustraciones

Ilustración 1: Documento	4
Ilustración 2: Esquema conceptual	5
Ilustración 3: Esquema Lógico(Estrella)	6
Ilustración 4: Copos de nieve	6
Ilustración 5: Esquema Físico en MySQL	7
Ilustración 6: Schema Workbench Inicial	7
Ilustración 7: Opciones Schema Workbench	8
Ilustración 8: Dimensiones Mondrian	8
Ilustración 9: Tabla de hechos en schema-workbench.....	9
Ilustración 10: Input de datos en PDI	10
Ilustración 11: Exportar datos CSV	11
Ilustración 12: Mapeo de campos.....	12
Ilustración 13: KTR con todas las tablas de dimensiones	12
Ilustración 14: Búsqueda en base de datos para la tabla de hechos.....	13
Ilustración 15: Mapping tabla hechos	14
Ilustración 16: ID tabla hechos	14
Ilustración 17: KTR Tabla Hechos	14
Ilustración 18: Login Pentaho Server	15
Ilustración 19: New Connection	15
Ilustración 20: PSW Publish Schema	16
Ilustración 21: Datos de publicación del cubo.....	16
Ilustración 22: New JPivot View	17
Ilustración 23: JPivot Schema & Cube	17
Ilustración 24: JPivot Cube Rollups.....	17
Ilustración 25: Obtención de datos PowerBI.....	18
Ilustración 26: Conexión a base de datos.....	18
Ilustración 27: Añadimos tablas a PowerBI	19
Ilustración 28: Carga de datos PowerBI	19
Ilustración 29: Administrar relaciones de la tabla de hechos.....	20
Ilustración 30: Relaciones: tabla árbitros	20
Ilustración 31: Todas las relaciones hechas.....	21
Ilustración 32: Visualizaciones en PowerBI	21
Ilustración 33: Todos el año 2017.....	22
Ilustración 34: Todos el año 2018.....	22
Ilustración 35: Ter Stegen 2018	23

1. Especificación de datos

Lo primero que tenemos que hacer es analizar el documento proporcionado por el cliente, reconociendo en dicho documento, las necesidades del mismo.

Inteligencia de Negocio y Gestión de Procesos (INGP)

Desarrolla un diseño lógico a partir de un Data Mart usando el esquema estrella. Identifica los atributos de cada tabla y determina los niveles de jerarquía que consideres para cada dimensión:

La organización de equipos de fútbol españoles ha decidido configurar un Data Warehouse para analizar las tendencias de los equipos y tratar de mejorar sus resultados. Para hacer esto, la administración del equipo quiere analizar la información disponible sobre los **partidos jugados en la liga**.

Los analistas quieren conocer datos relevantes tanto sobre partidos como sobre jugadores. Para cada **partido** quieren saber los **goles marcados por cada jugador** y los **goles que ha evitado**. También quieren saber información sobre las **tarjetas amarillas** y **rojas** recibidas por cada jugador en un partido determinado. Finalmente, quieren saber el **tiempo** que el jugador ha pasado en el campo.

Además de la información sobre el rendimiento de los **jugadores**, los analistas quieren recopilar otros datos, como el **nombre** del jugador, su **posición**, **edad** y **estatura**. Asimismo, quieren agrupar a los jugadores en **equipos** para analizar el rendimiento de los diferentes equipos en la temporada. Del **equipo** se conoce su **nombre** y el **valor en euros** del equipo.

Por otro lado, quieren conocer información sobre el **estadio del partido** con el fin de analizar si la ubicación influye en el resultado. El estadio tiene una **capacidad máxima**, **nombre**, **calidad del terreno de juego**, **año de inauguración**, **ciudad** y **país** desde donde se encuentra.

Además de estos datos, un entrenador ha indicado que está interesado en conocer datos sobre los **árbitros** que supervisan cada partido para poder reclamar un arbitraje justo. Para ello, se necesita conocer el **nombre** del árbitro, su **edad** y su **país de origen**.

Para finalizar, también se ha decidido almacenar la **fecha del partido**, incluida la información sobre el **día**, el **mes** y el **año** en que se celebra y el **clima** durante la celebración.

Ilustración 1: Documento

Como podemos ver, dividimos el documento en 2 tipos de datos:

- El **hecho(cubo)** subrayado en rojo, con sus **atributos**.
- Las **dimensiones** subrayadas en verde, con sus **atributos**.

Ahora vamos a crear el esquema conceptual usando el esquema estrella, como nos indica en el título.

2. Esquema Conceptual

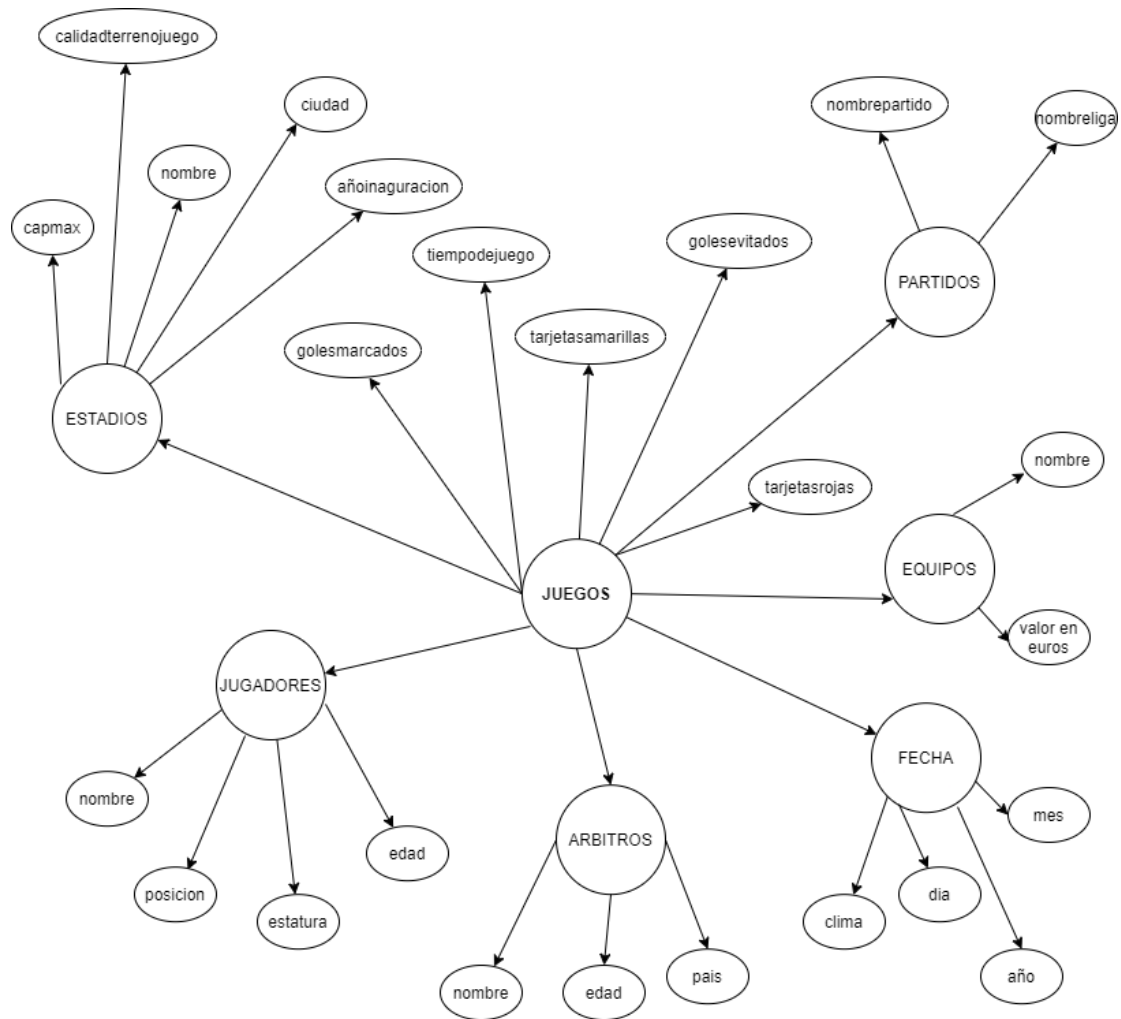


Ilustración 2: Esquema conceptual

2.1 Diferencias entre esquema estrella y copos de nieve

¿Por qué hemos usado este diseño en vez de copos de nieve o constelaciones?

Sencillo. Cuando analizamos el documento nos aparecen atributos por tabla y en el caso de la tabla **estadios** hay dos atributos que se podría contemplar jerarquizar en cadena, es decir, un país tiene ciudades y las ciudades tienen estadios.

Si se diera este caso, o el cliente nos aportara datos más adelante y nos especificara que necesita esa información en distintas tablas, esa parte SÍ sería tipo copos de nieve.

A continuación, vemos la diferencia entre ambos en el diseño lógico.

3. Esquema Lógico

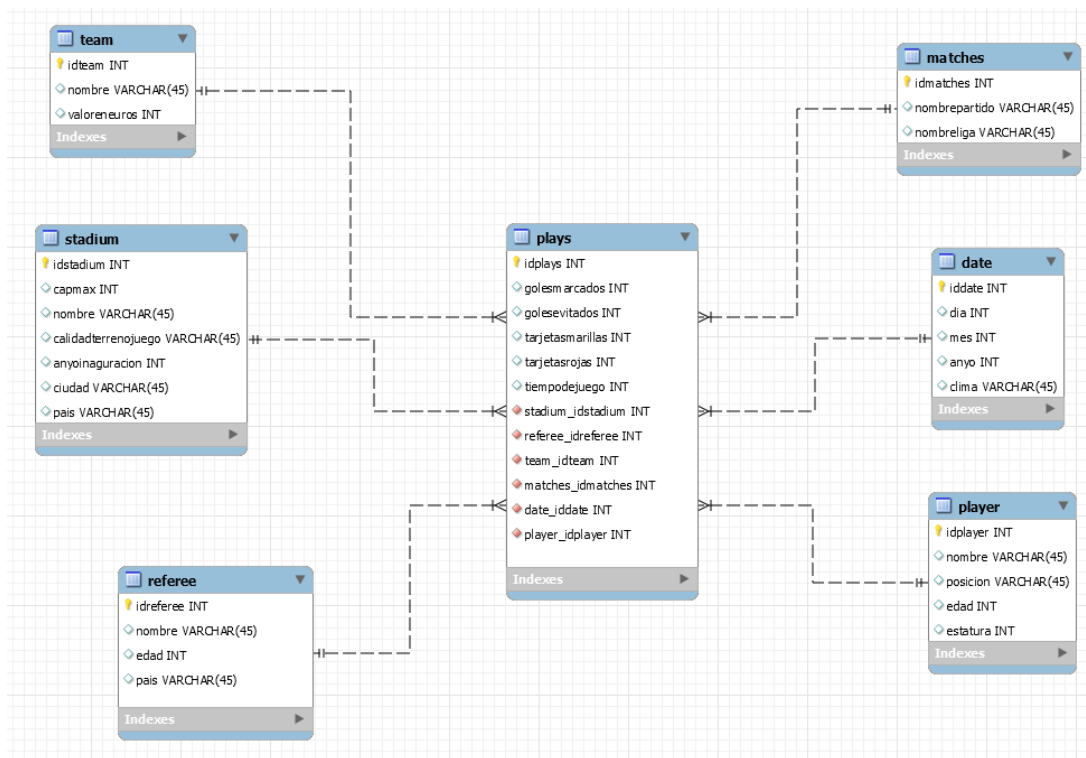


Ilustración 3: Esquema Lógico(Estrella)

Como podemos observar, el esquema es tipo estrella, puesto que tenemos una tabla central que contempla todas las medidas y FK de las demás tablas.

Abajo tenemos la posibilidad de que lo comentado antes se diera. Este tipo de esquema, se llama copos de nieve, donde la jerarquía va en cadena.

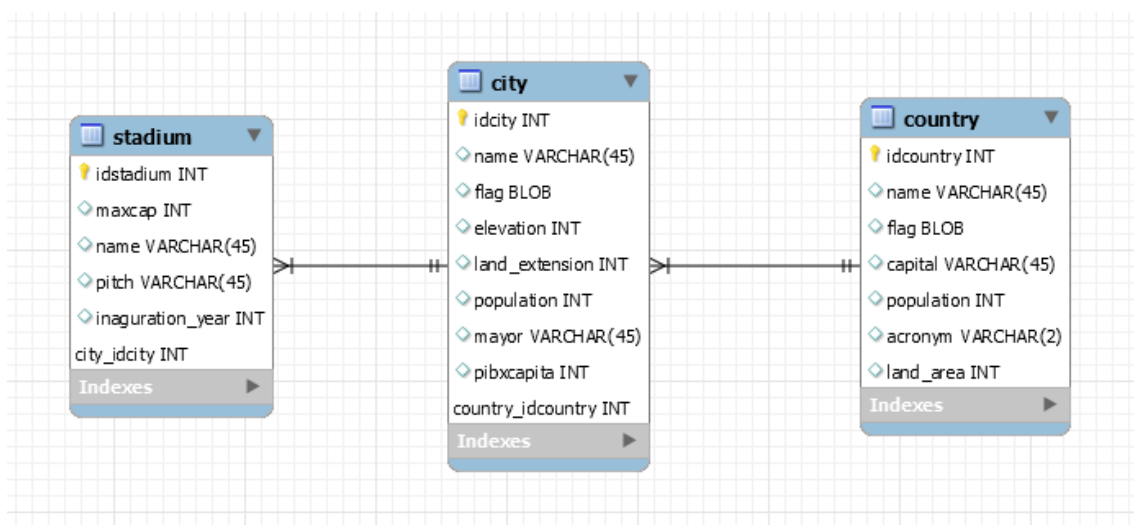
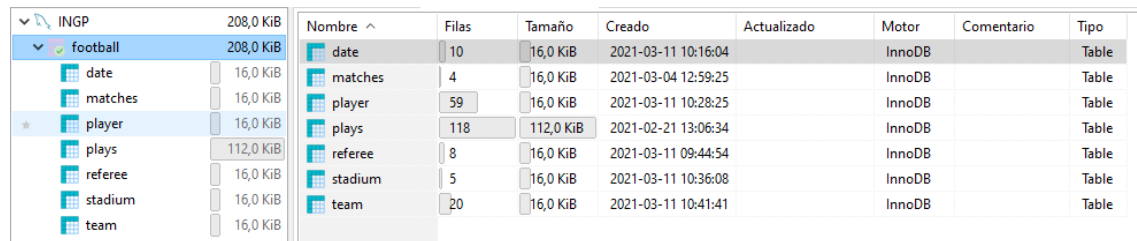


Ilustración 4: Copos de nieve

4. Esquema Físico

Una vez realizado el esquema lógico en el creador de diagramas EER en MySQL Workbench, pulsamos en la opción *Database* → *Forward Engineer*.

Si el proceso es correcto, tendremos en nuestra base de datos el esquema lógico transformado a tablas.



Nombre	Filas	Tamaño	Creado	Actualizado	Motor	Comentario	Tipo
date	10	16,0 KiB	2021-03-11 10:16:04		InnoDB		Table
matches	4	16,0 KiB	2021-03-04 12:59:25		InnoDB		Table
player	59	16,0 KiB	2021-03-11 10:28:25		InnoDB		Table
plays	118	112,0 KiB	2021-02-21 13:06:34		InnoDB		Table
referee	8	16,0 KiB	2021-03-11 09:44:54		InnoDB		Table
stadium	5	16,0 KiB	2021-03-11 10:36:08		InnoDB		Table
team	20	16,0 KiB	2021-03-11 10:41:41		InnoDB		Table

Ilustración 5: Esquema Físico en MySQL

5. Pentaho Schema Workbench

Una vez tenemos creada la estructura de la base de datos, pero sin tener datos insertados, es hora de crear el archivo XML que realice el mapeo de datos para definir así, una estructura de base de datos multidimensional.

Para ello usaremos el software **Pentaho Schema Workbench**

Después de crear la variable del sistema que permita ejecutar la JVM que soporte este software y pulsando en *workbench.bat*, nos aparecerá esta ventana:



Ilustración 6: Schema Workbench Inicial

Una vez dentro, pulsaremos en *New* → *Schema*

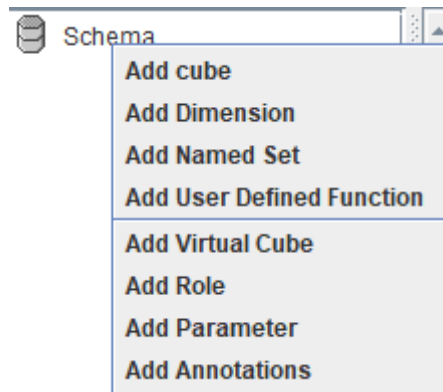


Ilustración 7: Opciones Schema Workbench

Iremos creando, fijándonos en el documento proporcionado por nuestro cliente, las dimensiones necesarias con sus atributos y al final crearemos el cubo, que corresponde a nuestra tabla de hechos.

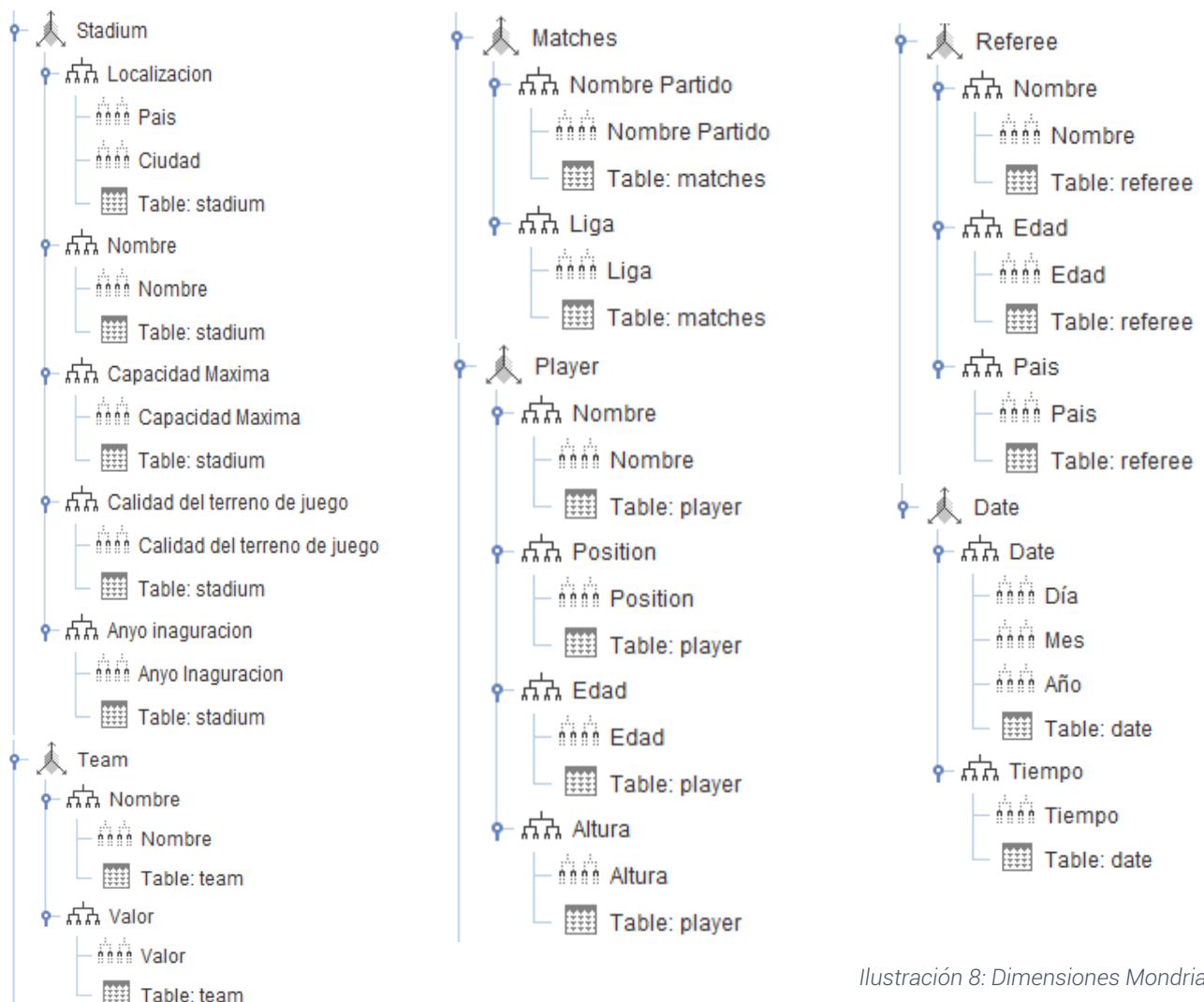






Ilustración 8: Dimensiones Mondrian

Arriba se puede ver como se realizan las dimensiones en el schema-workbench:

-  → DIMENSIÓN: Contiene las dimensiones del esquema lógico multidimensional
-  → JERARQUÍAS: Contiene las tablas y los atributos jerarquizados de cada dimensión
-  → NIVELES: Contiene los niveles de atributos de cada jerarquía.
 - En el caso de *stadium*, podemos ver que el nivel *Localización* contiene 2 atributos, ciudad y país. La lógica de este agrupamiento es aplastante, puesto que una ciudad y un país son localizaciones.
 - La fecha se divide en *año*, *mes* y *día*, **PERO** todos están al mismo nivel, puesto que una fecha se puede descomponer en estos 3 atributos.
-  → TABLAS: Es necesario asignar a cada dimensión de que tabla proviene.

Por último, generamos el cubo (tabla de hechos)

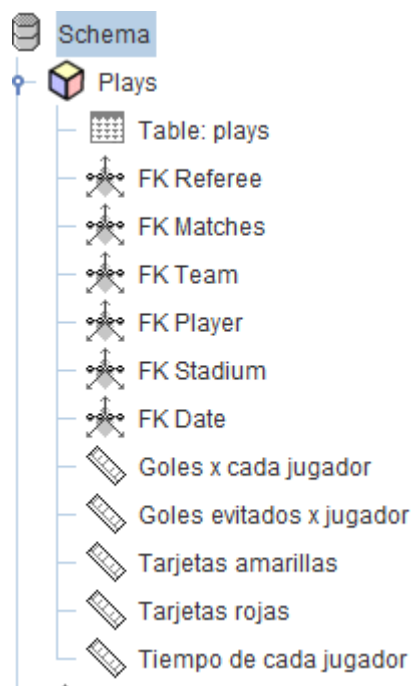






Ilustración 9: Tabla de hechos en schema-workbench

Éste a su vez se divide en:

-  → CUBO: Contiene la tabla de hechos
-  → TABLA: Tabla de la base de datos (plays)
-  → DIMENSION USAGE: Nexos entre las dimensiones y el cubo
-  → MEDIDAS: Atributos a medir de la tabla de hechos

6. PDI (Pentaho Data Integration)

Una vez realizado el esquema multidimensional, pasamos a poblar nuestra base de datos con los CSV aportados por nuestro cliente.

Para ello, vamos a usar el software también de Pentaho llamado data-integration que sirve para ingerir, combinar, limpiar y preparar diversos datos de cualquier fuente en cualquier entorno.

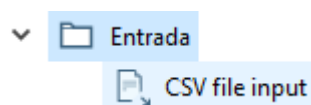
En nuestro caso, como ya he comentado, necesitamos ingerir y transformar los datos de los CSV.

Para ello, una vez iniciado el software(*spoon.bat*) nos fijamos en la pestaña diseño que tenemos la posibilidad de *Transformar*.

Una vez dentro, se nos presentan varias carpetas, nosotros vamos a usar solo por ahora, la de entrada y salida.

6.1 Entrada

NOTA: Como el proceso para nutrir a nuestra base de datos con las dimensiones son **iguales** lo haremos solo en el caso de la dimensión **ARBITROS**



Creamos un tipo de *CSV file input* y pulsamos doble sobre este elemento.

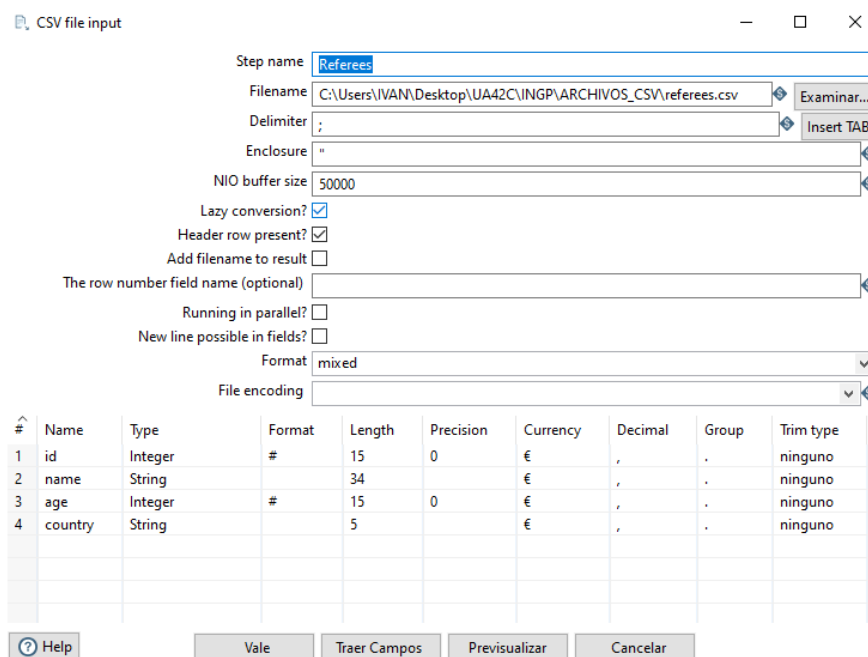
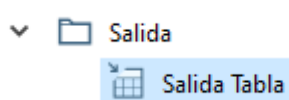


Ilustración 10: Input de datos en PDI

Voy a explicar campo a campo lo que necesitamos para importar nuestro CSV:

- Step name: Nombre simbólico para este paso
- Filename: Examinamos y buscamos nuestro CSV de árbitros
- Delimiter: En nuestro caso, los CSV proporcionado por nuestro cliente tienen un punto y coma como delimitador, lo especificamos en este campo.
- Traer campos: Nos permite visualizar los campos recuperados de nuestro CSV
- Previsualizar(Opcional): Nos muestra en una ventana los datos obtenidos en columnas.

6.2 Salida



Creamos un tipo de *Salida tabla* y pulsamos doble sobre este elemento.

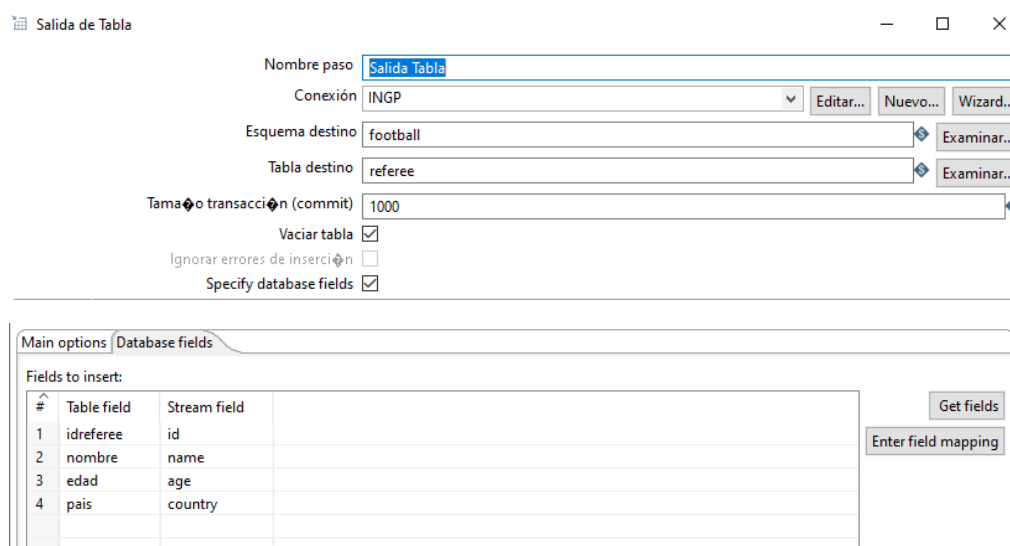


Ilustración 11: Exportar datos CSV

Voy a explicar campo a campo lo que necesitamos para exportar nuestro CSV a una tabla en MySQL:

- Nombre paso: Nombre simbólico
- Conexión: Creamos la conexión con nuestra base de datos tipo MySQL y JDBC con todos los datos que nos piden.
- Esquema destino: El esquema al que va relacionado (no rellenar)
- Tabla destino: La tabla **árbitros** de nuestra base de datos. Al rellenar esta tabla se rellena el esquema de arriba (el paso anterior)
- Vaciar tabla: Antes de introducir los datos vacía la tabla de MySQL.
- Specify database fields: Al marcar este checkbox, se nos desbloquea la opción de obtener y mapear los campos en nuestro proceso ETL.

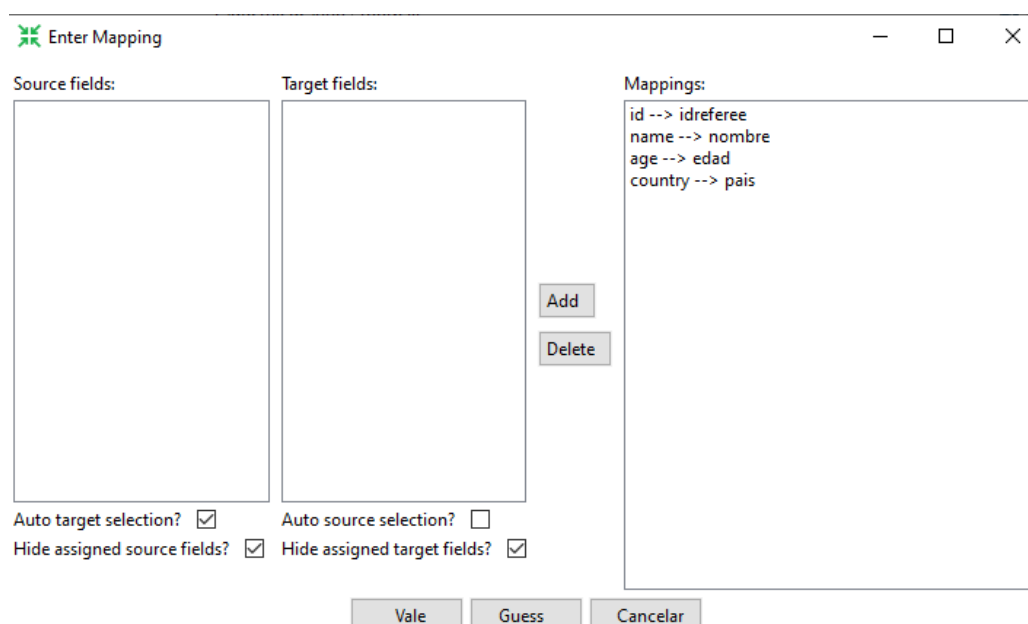


Ilustración 12: Mapeo de campos

Una vez mapeados los campos, aceptamos y pulsamos en ejecutar, en la parte superior.

Nos pedirá guardar nuestro KTR y a continuación lo ejecutará.

Si todo va bien, nos debe aparecer tanto la importación de datos como la exportación a MySQL con un tick en verde.

A continuación, podemos ver la inserción de todas nuestras tablas con todos los procesos ETL en un solo KTR. Cabe destacar que este paso es recomendable hacerlo al final, cuando cada KTR por separado funcione, ya que, si uno falla, ya no continua la ejecución, aun siendo posible que las posteriores funcionaran.

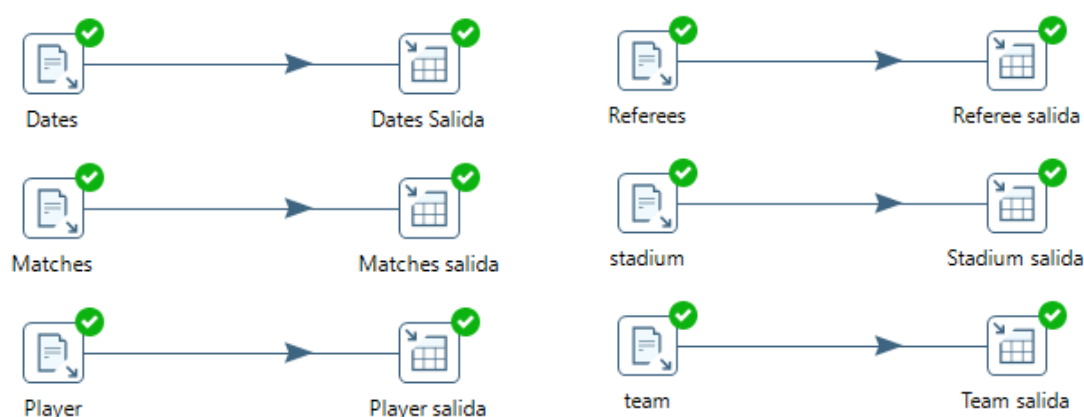


Ilustración 13: KTR con todas las tablas de dimensiones

6.3 Tabla de hechos

Para nuestra tabla de hechos el esquema es algo distinto:

Para empezar, añadimos una entrada de datos, y lo configuramos como anteriormente.

Pero ahora añadimos *Búsqueda* → *Búsqueda en base de datos*, que nos permitirá relacionar las claves ajenas.

Nombre paso: Búsqueda en base de datos matches

Conexión: INGP

Esquema de búsqueda: football

Tabla de búsqueda: matches

Habilitar cache? ☐

Tamaño de cache en filas (0=todas): 0

Load all data from table ☐

La clave(s) para realizar búsqueda de valor(es):

#	Campo de tabla	Comparador	Campo1	Campo2
1	name	=	match	

Valores a devolver de la tabla de búsqueda:

#	Campo	Nuevo nombre	Defecto	Tipo
1	id			Integer

No procesar la fila si la búsqueda falla ☐

Producir error si se obtienen multiples ☐

Ordenar por:

Buttons: Help, Vale, Cancelar, Obtener Campos, Obtener Campos Búsqueda

Ilustración 14: Búsqueda en base de datos para la tabla de hechos

Voy a explicar campo a campo:

- Nombre paso: nombre simbólico del paso
- Conexión: La conexión a nuestra base de datos
- Esquema de búsqueda: No rellenar, se hará automáticamente
- Tabla de búsqueda: Tabla de nuestro esquema al que queremos relacionar con nuestra tabla de hechos
- La clave(s) para realizar la búsqueda de valor(es): Este grid nos permite relacionar los datos, es decir, si en nuestro CSV de nuestra tabla de hechos tenemos el nombre del partido, necesitamos compararlo con el nombre, **pero** de la tabla de partidos.
- Valores a devolver de la tabla de búsqueda: Una vez hecha la relación, este grid nos indica que valor de la tabla partidos debe devolver para que la *foreign key* funcione con la tabla de hechos, en este y los demás casos es el id.

Debemos ir encadenando *búsquedas* de **todas las tablas de dimensiones** hasta llegar a la salida, que configuraremos como antes **EXCEPTO** por un paso:

En el CSV, no tenemos el id, por lo que el mapping nos quedaría así:

Enter Mapping

Source fields:	Target fields:
player	idplays
referee	
date	
stadium	
match	
team	

Ilustración 15: Mapping tabla hechos

¿Y qué hacemos con el id?

Volvemos atrás y en la ventana de *Salida* tenemos la pestaña “Main options” y la siguiente opción:

Incluye clave auto-generada ☒

Nombre del campo clave auto-generada

Ilustración 16: ID tabla hechos

Esto indica a Pentaho data integration que debe introducir un id autogenerado para que nuestro KTR funcione.

Ejecutamos y obtenemos el siguiente resultado:

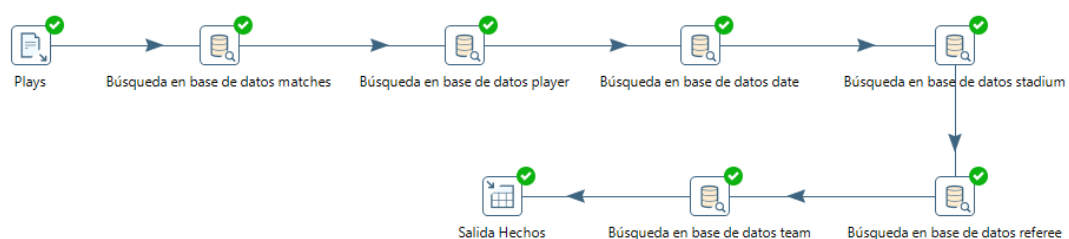


Ilustración 17: KTR Tabla Hechos

El paso siguiente, sería comprobar que nuestra base de datos está poblada, este paso no voy a mostrarlo puesto que solamente sería hacer SELECT's en nuestra base de datos.

7. Pentaho Server

El software Pentaho Server nos permite visualizar en formato JPivot nuestro esquema multidimensional y poder hacer rollups en cada atributo:

Iniciamos nuestro server(*start-pentaho.bat*) lo que ejecuta un proceso que levanta un servidor Tomcat, el cuál soportará nuestro server.

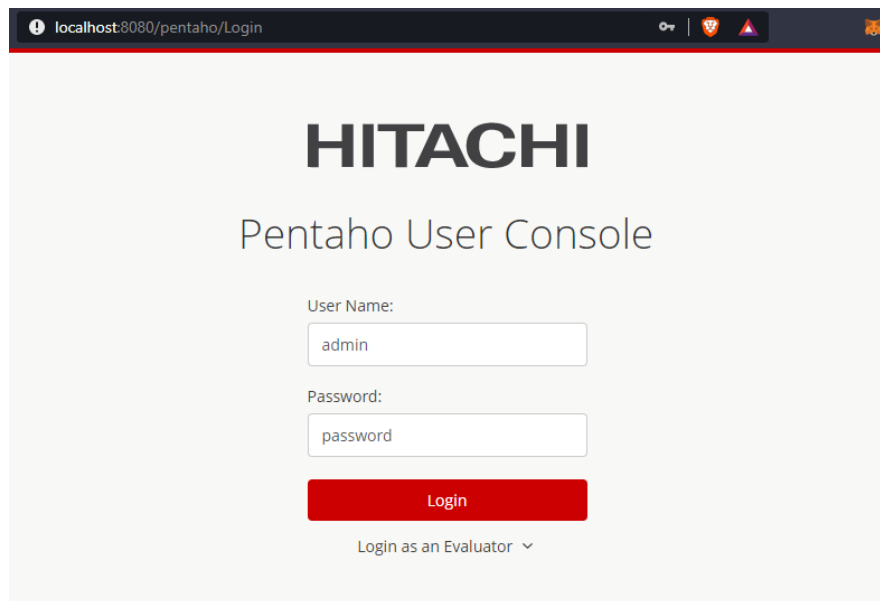


Ilustración 18: Login Pentaho Server

Iniciamos sesión y configuramos una conexión a nuestra base de datos entrando a la opción *Manage Data Sources*:

Database Connection

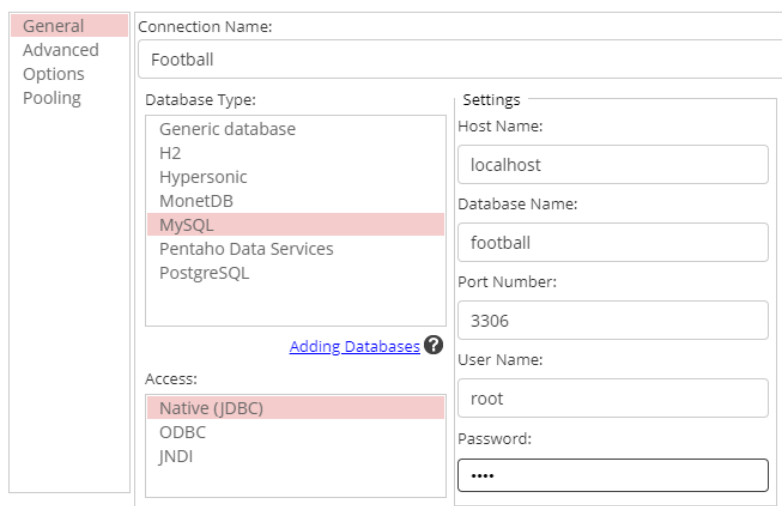


Ilustración 19: New Connection

A continuación, entramos al Schema Workbench y *Publicamos* nuestro esquema:

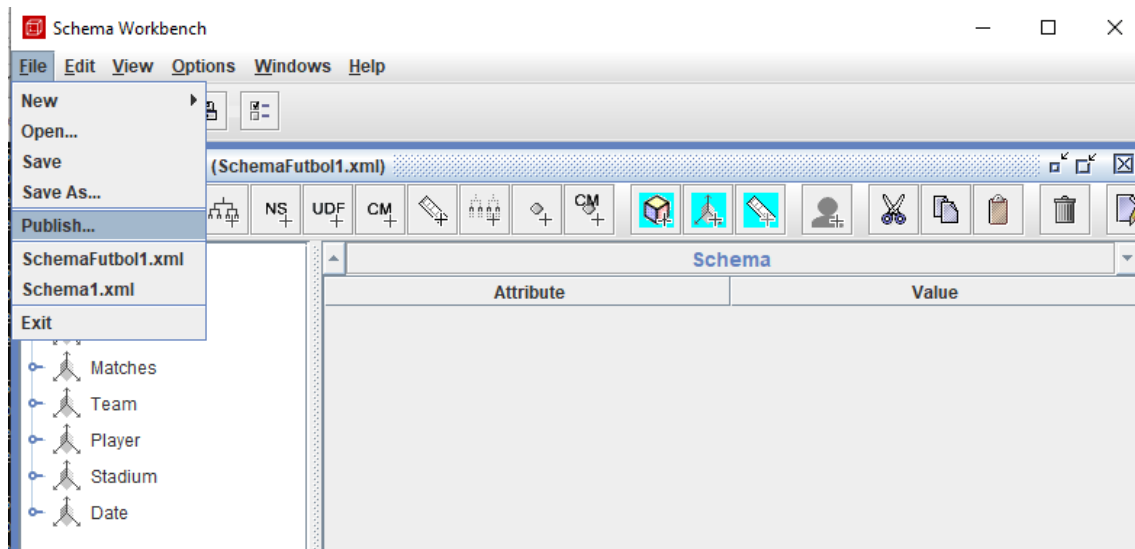


Ilustración 20: PSW Publish Schema

En la siguiente ventana, completamos los campos con los datos necesarios para acceder a nuestro server:

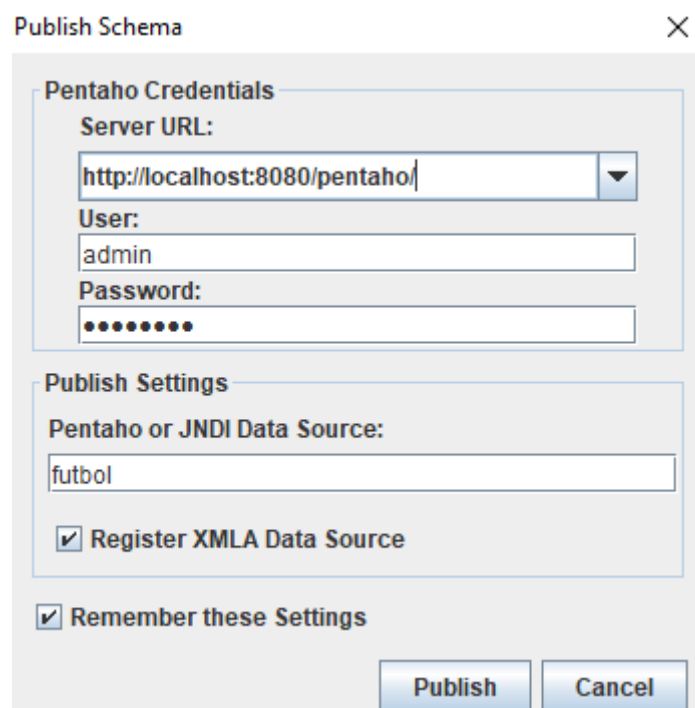


Ilustración 21: Datos de publicación del cubo

Ahora en Pentaho Server, creamos una vista JPivot

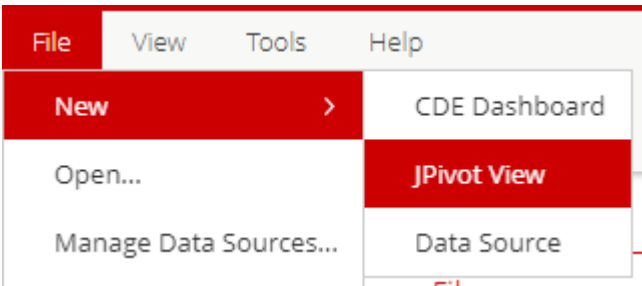


Ilustración 22: New JPivot View

Seleccionamos nuestro esquema y el cubo:

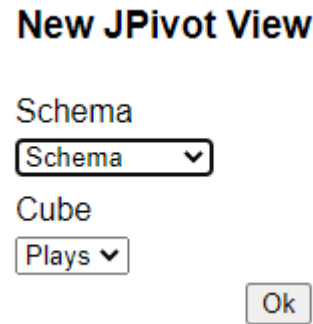


Ilustración 23: JPivot Schema & Cube

Y se nos mostrará una vista para hacer rollups con cada atributo:

						Medidas
FK Referee.Nombre	FK Matches.Nombre Partido	FK Team.Nombre	FK Player.Nombre	FK Stadium.Localizacion	FK Date.Date	Goles x cada jugador
All Referee.Nombres	All Matches.Nombre Partidos	All Team.Nombres	All Player.Nombres	All Stadium.Localizacions	All Dates	12
					1	2
					4	2
					7	3
					22	5
Inmaculada Prieto Martinez	All Matches.Nombre Partidos	All Team.Nombres	All Player.Nombres	All Stadium.Localizacions	All Dates	3
					7	3
Mario Melero Lopez	All Matches.Nombre Partidos	All Team.Nombres	All Player.Nombres	All Stadium.Localizacions	All Dates	2
Paola Cebollada Lopez	All Matches.Nombre Partidos	All Team.Nombres	All Player.Nombres	All Stadium.Localizacions	All Dates	2
					1	2
					4	2
Santiago Jaime Latre	All Matches.Nombre Partidos	All Team.Nombres	All Player.Nombres	All Stadium.Localizacions	6	2
					2018	2
					All Dates	5
					22	5

Ilustración 24: JPivot Cube Rollups

Como podemos ver, tenemos las distintas FK de nuestro cubo y podemos ir desplegando los campos.

8. PowerBI

Ahora vamos a analizar los datos con el software de Microsoft PowerBI que año a año, se ha ido ganando la corona como software para análisis de datos.

Entramos y seleccionamos la opción *Obtener datos* → *Base de datos* → *MySQL*

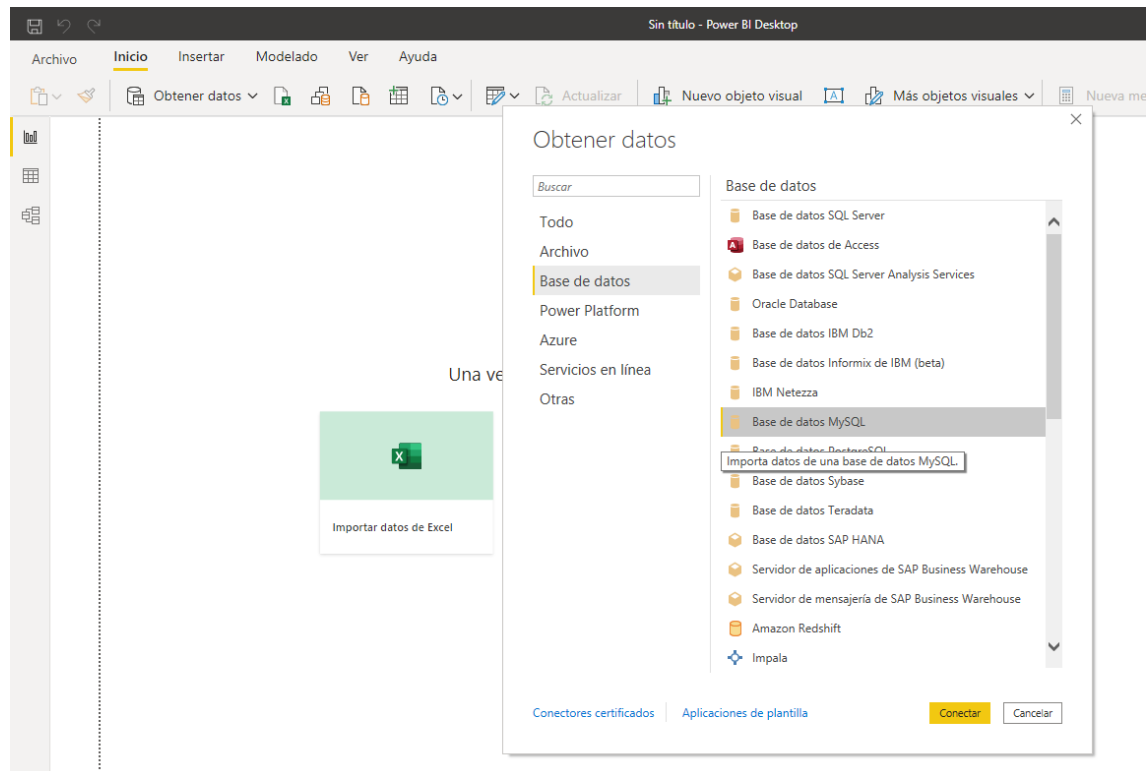


Ilustración 25: Obtención de datos PowerBI

A continuación, añadimos los datos de conexión y éste, nos mostrará las tablas a posteriori

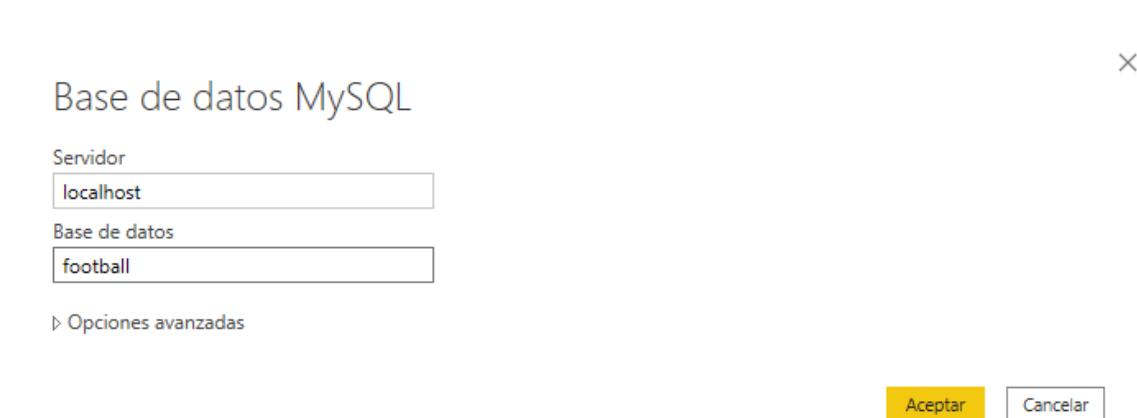


Ilustración 26: Conexión a base de datos

Seleccionamos todas las tablas y le damos a *Cargar*

Navegador

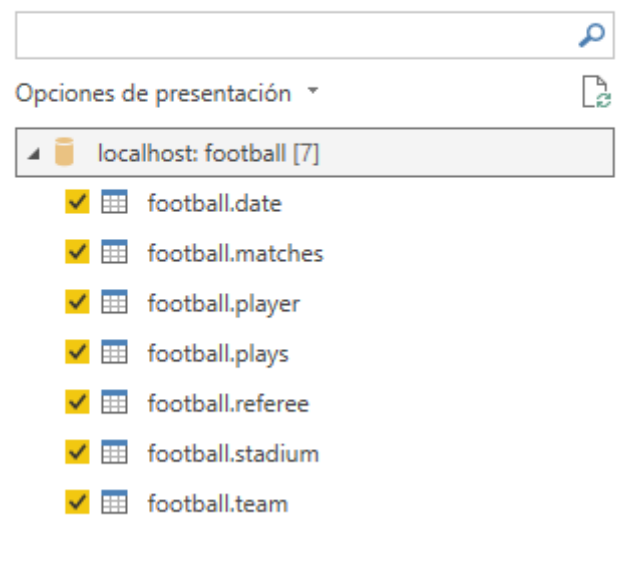


Ilustración 27: Añadimos tablas a PowerBI

PowerBI empezará a cargar todas las tablas

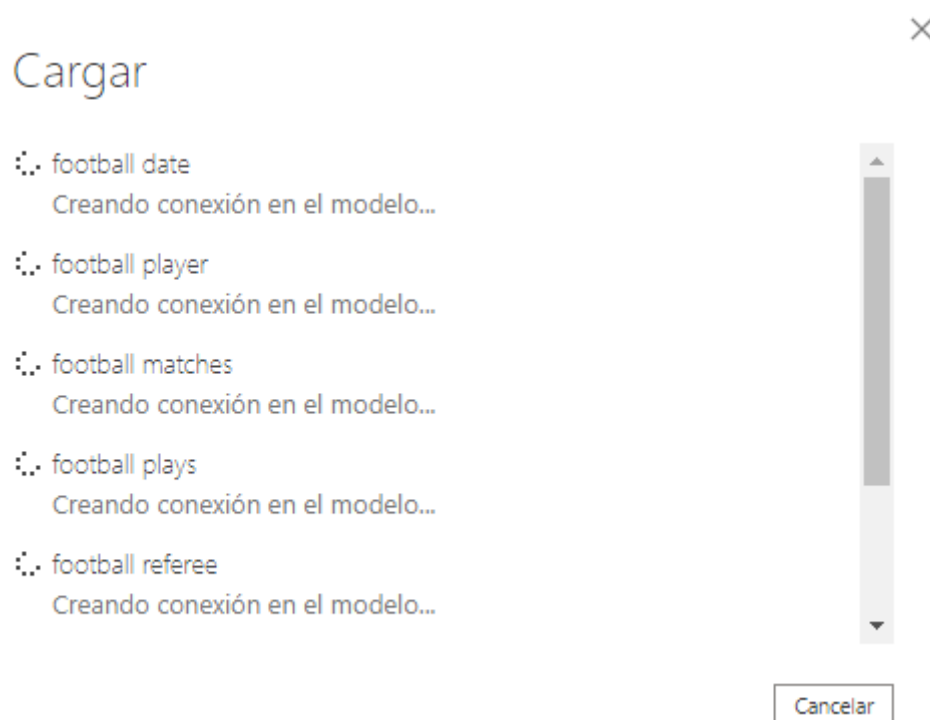


Ilustración 28: Carga de datos PowerBI

Una vez cargados los datos, tenemos que *Administrar relaciones* de la tabla de hechos:

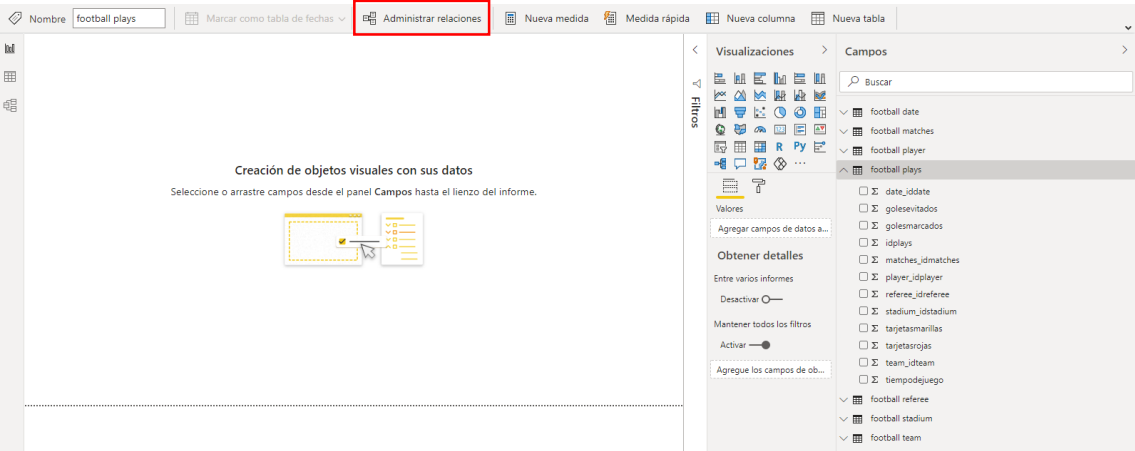


Ilustración 29: Administrar relaciones de la tabla de hechos

Creamos las relaciones de esta tabla con todas las demás de la siguiente forma:

Crear relación

Permite seleccionar tablas y columnas relacionadas.

football plays

stadium_idstadium	referee_idreferee	team_idteam	matches_idmatches	date_iddate	player_idplayer
1	5	4	1	4	3
1	5	4	1	4	5
1	5	4	1	4	6

football referee

idreferee	nombre	edad	pais
1	Mario Melero Lopez	39	Spain
2	Jose Luis Munuera Montero	35	Spain
3	Inmaculada Prieto Martinez	25	Spain

Cardinalidad

Varios a uno (*:1)

Dirección del filtro cruzado

Única

☒ Activar esta relación

☐ Aplicar filtro de seguridad en ambas direcciones

☐ Asumir integridad referencial

Aceptar

Cancelar

Ilustración 30: Relaciones: tabla árbitros

Administrar relaciones

Activo	Desde: tabla (columna)	A: tabla (columna)
<input checked="" type="checkbox"/>	football plays (date_iddate)	football date (iddate)
<input checked="" type="checkbox"/>	football plays (matches_idmatches)	football matches (id)
<input checked="" type="checkbox"/>	football plays (player_idplayer)	football player (idplayer)
<input checked="" type="checkbox"/>	football plays (referee_idreferee)	football referee (idreferee)
<input checked="" type="checkbox"/>	football plays (stadium_idstadium)	football stadium (idstadium)
<input checked="" type="checkbox"/>	football plays (team_idteam)	football team (idteam)

Ilustración 31: Todas las relaciones hechas

Y a continuación vamos añadiendo formas de visualizar dichos datos:



Ilustración 32: Visualizaciones en PowerBI

Añadimos 4 del tipo Segmentación de datos y 2 Gráfico de columnas agrupadas

El resultado es el siguiente:

El nombre del jugador y el día, mes y año como segmentación y las 2 gráficas que dividen los goles y las tarjetas:

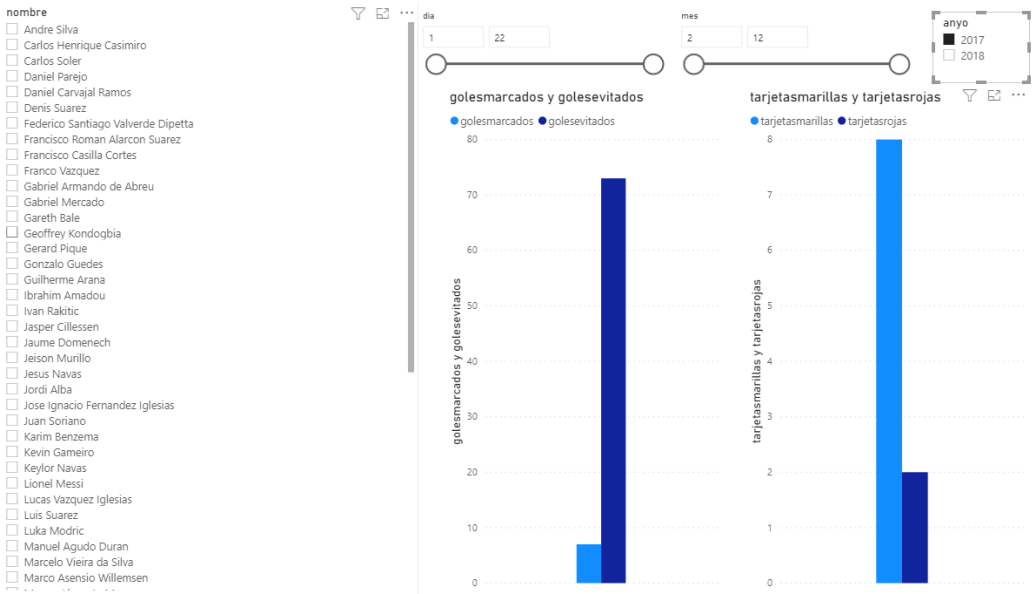


Ilustración 33: Todos el año 2017

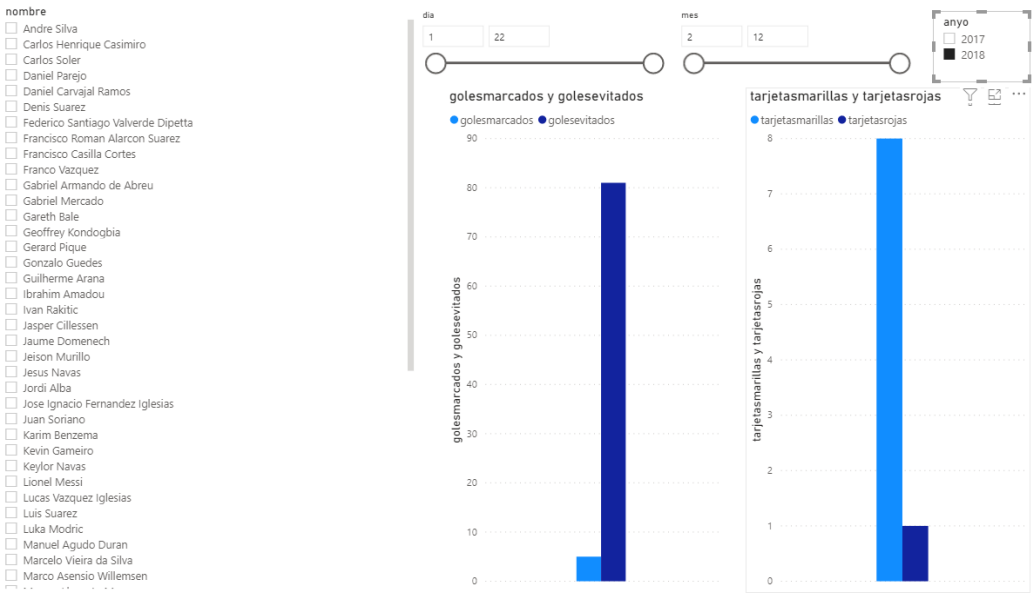


Ilustración 34: Todos el año 2018

Es interesante ver como hay muchos más goles evitados en 2018 que en 2017.

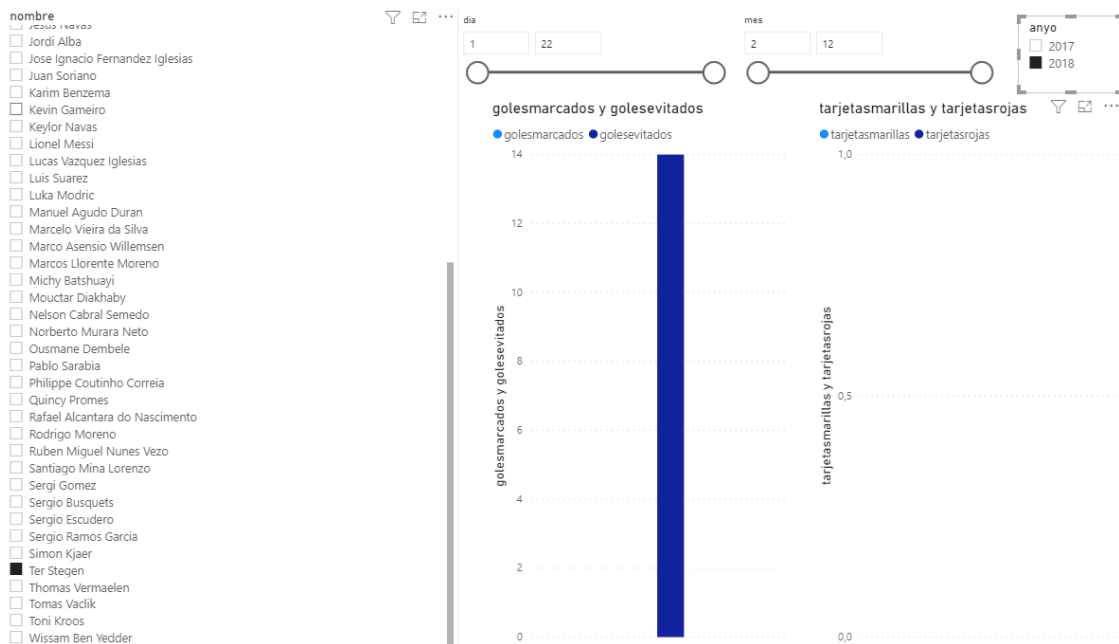


Ilustración 35: Ter Stegen 2018

Las estadísticas de un portero, como es normal, no tiene goles marcados, muchos evitados y en este caso, Ter Stegen no tiene tarjetas ni amarillas ni rojas para el año 2018 en nuestro dataset.

9. Bibliografía

MySQL

<https://dev.mysql.com/downloads/workbench/>

MySQL connector for PowerBI

<https://dev.mysql.com/downloads/connector/net/>

PDI

<https://sourceforge.net/projects/pentaho/files/latest/download>

Pentaho Workbench

<https://sourceforge.net/projects/pentaho/files/Pentaho%209.0/client-tools/psw-ce-9.0.0.0-423.zip/download>

Pentaho Server

https://sourceforge.net/projects/pentaho/files/Pentaho%208.2/server/pentaho-server-ce-8.2.0.0-342.zip/download?use_mirror=deac-ams&download=&failedmirror=deac-riga.dl.sourceforge.net

PowerBI Desktop

<https://www.microsoft.com/en-us/download/details.aspx?id=58494>