

APLICACIÓN DE CRITERIOS DE DISPONIBILIDAD DE BASES DE DATOS DISTRIBUIDAS Y REPLICADAS

Raúl Arribas Redondo

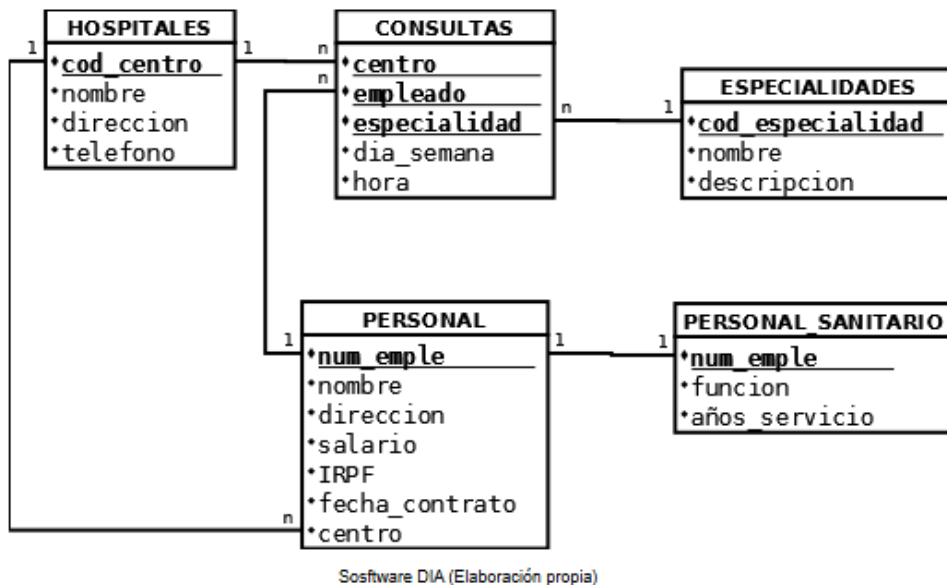
BBDD

1 Contenido

2	Fragmentación Vertical.....	3
2.1	Código tablas consultas :	3
3	Fragmentación Vertical:.....	5
3.1	Código tabla personal	5
4	Beneficios de hacer las fragmentaciones	6
5	Código completo de MYSQL	6
6	Modelo Diagrama	9

El Servicio Cántabro de Salud cuenta con tres hospitales comarcales, uno en Santander, otro en Laredo y el tercero en Torrelavega. Cada uno de ellos cubre determinados servicios médicos delimitados por especialidades, pudiéndose atender una especialidad en más de un hospital. Cada hospital tiene, al menos, una especialidad.

Ahora mismo todo está en una base de datos centralizada, que sigue el siguiente esquema:



Software DIA (Elaboración propia)

Se quiere distribuir la BD sabiendo que:

1. En la sede central, el hospital de Santander, se elaboran las nóminas de todo el personal que trabaja en los hospitales comarcales.
2. Se desea dar autonomía local a cada hospital para gestionar el personal sanitario que trabaja en el mismo, así como el horario de las consultas y la especialidad que se atiende.

Tu **tarea consiste** en realizar **el esquema de fragmentación y la asignación** de los ficheros de la BDD. Para ello debes redactar un documento donde se vean claramente las **acciones del esquema de fragmentación y la asignación de los ficheros de la BD**.

2 Fragmentación Vertical

En primer lugar vamos a realizar una fragmentación horizontal (****La fragmentación horizontal implica dividir una tabla en varias subtablas basándose en ciertos criterios****) de la tabla Consulta, la vamos a dividir en 3 tablas mas con los siguientes criterios:

- **CONSULTAS_SANTANDER:** Contiene todas las consultas realizadas en el Hospital de Santander.
- **CONSULTAS_LAREDO:** Contiene todas las consultas realizadas en el Hospital de Laredo.
- **CONSULTAS_TORRELAVEGA:** Contiene todas las consultas realizadas en el Hospital de Torrelavega.

Cada una de estas tablas tiene la misma estructura que la tabla **CONSULTAS** original, pero con datos específicos para cada hospital.

2.1 Código tablas consultas:

-- Tabla CONSULTAS_LAREDO

```
CREATE TABLE IF NOT EXISTS consultas_laredo (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    centro INT,  
    empleado INT,  
    especialidad INT,  
    dia_semana VARCHAR(10),  
    hora TIME,  
    FOREIGN KEY (centro) REFERENCES hospitales (cod_centro),  
    FOREIGN KEY (empleado) REFERENCES personal_info (num_emple),  
    FOREIGN KEY (especialidad) REFERENCES especialidades (cod_especialidad)  
);
```

-- Tabla CONSULTAS_SANTANDER

```
CREATE TABLE IF NOT EXISTS consultas_santander (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    centro INT,  
    empleado INT,  
    especialidad INT,  
    dia_semana VARCHAR(10),  
    hora TIME,
```

```
FOREIGN KEY (centro) REFERENCES hospitales (cod_centro),  
FOREIGN KEY (empleado) REFERENCES personal_info (num_emple),  
FOREIGN KEY (especialidad) REFERENCES especialidades (cod_especialidad)  
);
```

-- Tabla CONSULTAS_TORRELAVEGA

```
CREATE TABLE IF NOT EXISTS consultas_torrelavega (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    centro INT,  
    empleado INT,  
    especialidad INT,  
    dia_semana VARCHAR(10),  
    hora TIME,  
    FOREIGN KEY (centro) REFERENCES hospitales (cod_centro),  
    FOREIGN KEY (empleado) REFERENCES personal_info (num_emple),  
    FOREIGN KEY (especialidad) REFERENCES especialidades (cod_especialidad)  
);
```

3 Fragmentación Vertical:

Ahora vamos a realizar una fragmentación Vertical (*La fragmentación vertical implica dividir una tabla en varias subtablas basándose en columnas. En este caso, hemos realizado una fragmentación.*) de la tabla personal, vamos a dividirlo en dos tablas, una que se llame **personal_info** y la otra **personal_nomina**.

- **PERSONAL_INFO:** Contiene información básica del personal como nombre, dirección, fecha de contrato y centro.
- **PERSONAL_NOMINA:** Contiene información de nómina como salario e IRPF.

Estas dos tablas están relacionadas por la columna num_emple, que es la clave primaria en ambas tablas.

3.1 Código tabla personal

-- Tabla PERSONAL_INFO

```
CREATE TABLE IF NOT EXISTS personal_info (  
    num_emple INT PRIMARY KEY,  
    nombre VARCHAR(100),  
    direccion VARCHAR(255),  
    fecha_contrato DATE,  
    centro INT,  
    FOREIGN KEY (centro) REFERENCES hospitales (cod_centro)  
);
```

-- Tabla PERSONAL_NOMINA

```
CREATE TABLE IF NOT EXISTS personal_nomina (  
    num_emple INT PRIMARY KEY,  
    salario DECIMAL(10, 2),  
    IRPF DECIMAL(5, 2),  
    FOREIGN KEY (num_emple) REFERENCES personal_info (num_emple)  
);
```

4 Beneficios de hacer las fragmentaciones

Optimizar rendimiento: Al dividir las tablas grandes en tablas mas pequeñas, estamos reduciendo el tiempo de respuesta y mejora el rendimiento general del sistema.

Autonomía Local: Cada hospital puede gestionarse sin depender de la sede central.

Centralización de funciones específicas: Por ejemplo la sede central, Santander, se puede centrar en llevar todo el tema de las nominas de todos los empleados de todos los hospitales.

Facilitar la gestión y el mantenimiento: Cada hospital solo puede acceder a los datos que le corresponden, con lo cual la gestión de los mismos es mas sencilla.

5 Código completo de MYSQL

```
CREATE DATABASE IF NOT EXISTS hospitales;
```

```
USE hospitales;
```

```
-- Tabla HOSPITALES
```

```
CREATE TABLE IF NOT EXISTS hospitales (
```

```
    cod_centro INT PRIMARY KEY,
```

```
    nombre VARCHAR(100),
```

```
    direccion VARCHAR(255),
```

```
    telefono VARCHAR(20)
```

```
);
```

```
-- Tabla PERSONAL_INFO
```

```
CREATE TABLE IF NOT EXISTS personal_info (
```

```
    num_emple INT PRIMARY KEY,
```

```
    nombre VARCHAR(100),
```

```
    direccion VARCHAR(255),
```

```
    fecha_contrato DATE,
```

```
    centro INT,
```

```
    FOREIGN KEY (centro) REFERENCES hospitales (cod_centro)
```

```
);
```

```
-- Tabla ESPECIALIDADES
```

```
CREATE TABLE IF NOT EXISTS especialidades (  
    cod_especialidad INT PRIMARY KEY,  
    nombre VARCHAR(100),  
    descripcion TEXT  
);
```

-- Tabla CONSULTAS_LAREDO

```
CREATE TABLE IF NOT EXISTS consultas_laredo (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    centro INT,  
    empleado INT,  
    especialidad INT,  
    dia_semana VARCHAR(10),  
    hora TIME,  
    FOREIGN KEY (centro) REFERENCES hospitales (cod_centro),  
    FOREIGN KEY (empleado) REFERENCES personal_info (num_emple),  
    FOREIGN KEY (especialidad) REFERENCES especialidades (cod_especialidad)  
);
```

-- Tabla CONSULTAS_SANTANDER

```
CREATE TABLE IF NOT EXISTS consultas_santander (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    centro INT,  
    empleado INT,  
    especialidad INT,  
    dia_semana VARCHAR(10),  
    hora TIME,  
    FOREIGN KEY (centro) REFERENCES hospitales (cod_centro),  
    FOREIGN KEY (empleado) REFERENCES personal_info (num_emple),  
    FOREIGN KEY (especialidad) REFERENCES especialidades (cod_especialidad)  
);
```

-- Tabla CONSULTAS_TORRELAVEGA

```
CREATE TABLE IF NOT EXISTS consultas_torrelavega (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    centro INT,  
    empleado INT,  
    especialidad INT,  
    dia_semana VARCHAR(10),  
    hora TIME,  
    FOREIGN KEY (centro) REFERENCES hospitales (cod_centro),  
    FOREIGN KEY (empleado) REFERENCES personal_info (num_emple),  
    FOREIGN KEY (especialidad) REFERENCES especialidades (cod_especialidad)  
);
```

-- Tabla PERSONAL_NOMINA

```
CREATE TABLE IF NOT EXISTS personal_nomina (  
    num_emple INT PRIMARY KEY,  
    salario DECIMAL(10, 2),  
    IRPF DECIMAL(5, 2),  
    FOREIGN KEY (num_emple) REFERENCES personal_info (num_emple)  
);
```

6 Modelo Diagrama

