Objetivos

Reforzar los conceptos básicos de Bases de Datos y adquirir autonomía

en el diseño del esquema de una base de datos relacional.

Practicar la recuperación de información mediante archivos de texto plano.

Jose Manuel Pinillos Rubio

BASES DE DATOS avanzadas

Diseño de una base de datos relacional

ÍNDICE

[ÍNDICE 2](#_Toc150776393)

[INTRODUCCIÓN 2](#_Toc150776394)

[DISEÑO DEL DIAGRAMA ENTIDAD-RELACIÓN 3](#_Toc150776395)

[Captura de requisitos 4](#_Toc150776396)

[Diseño conceptual 4](#_Toc150776397)

[**Entidades** 4](#_Toc150776398)

[**Atributos** 4](#_Toc150776399)

[**Relaciones** 6](#_Toc150776400)

[**Relaciones entre tablas** 6](#_Toc150776401)

[**Claves primarias y foráneas** 6](#_Toc150776402)

[Diagrama Entidad-Relación 7](#_Toc150776403)

[Explicación de las relaciones 8](#_Toc150776404)

[**Relación "pertenece" entre "municipalities" y "provinces"** 8](#_Toc150776405)

[**Relación "pertenece" entre "localities" y " municipalities"** 8](#_Toc150776406)

[**Relación "gestiona" entre "operators" y " stations"** 8](#_Toc150776407)

[**Relación "tiene" entre "stations" y " fuels"** 9](#_Toc150776408)

[Diseño lógico 10](#_Toc150776409)

[CONSULTAS 11](#_Toc150776410)

[**1. Nombre de la empresa con más estaciones de servicio terrestres** 11](#_Toc150776411)

[**2. Nombre de la empresa con más estaciones de servicio marítimas** 11](#_Toc150776412)

[**3. Localización, nombre de empresa y margen de la estación con el precio más bajo para el combustible «Gasolina 95 E5» en la Comunidad de Madrid** 11](#_Toc150776413)

[**4. Localización, nombre de empresa y margen de la estación con el precio más bajo para el combustible «Gasóleo A» si resido en el centro de Albacete y no quiero desplazarme más de 10 KM** 11](#_Toc150776414)

[**5. Provincia en la que se encuentre la estación de servicio marítima con el combustible «Gasolina 95 E5» más caro** 13](#_Toc150776415)

[CONCLUSIONES 14](#_Toc150776416)

**INTRODUCCIÓN**

La gestión de bases de datos en SQL es un componente fundamental en el ámbito de la Ingeniería Informática. En este trabajo, se aborda el diseño y la implementación de un sistema de bases de datos utilizando el enfoque del diagrama Entidad-Relación y su transformación en tablas relacionales. Además, se realizará la recuperación de información mediante archivos de texto plano, en este caso CSV. El objetivo principal será aplicar los conocimientos adquiridos en esta área mediante el diseño de una base de datos relacional, a partir de los datos de precios de combustibles en las diferentes comunidades autónomas de España y su posterior recuperación a nuestra base de datos.

El proceso se ha llevado a cabo siguiendo un orden estructurado. En primer lugar, se descargaron los datos con los que se trabajarían, para después realizar una limpieza de estos y exportarlos en un formato de texto plano.

Posteriormente, se realizó el diseñó el diagrama Entidad-Relación, que permitió representar las entidades, atributos y relaciones presentes del dominio de la base de datos. Se ha puesto especial énfasis en comprender y aplicar las cardinalidades, tanto para establecer las relaciones fuertes como para identificar las relaciones débiles.

A continuación, se procedió a la transformación del diagrama Entidad-Relación en un modelo relacional mediante un script SQL. Este proceso implica la creación de tablas que reflejen adecuadamente la estructura y las restricciones de la base de datos, garantizando la integridad y consistencia de los datos.

Posteriormente, se implementó el código de JDBC para la inserción de los datos correspondientes a la base de datos. Esta fase fue crucial para poblar la base de datos y permitir la realización de consultas y análisis posteriores.

Finalmente, se llevaron a cabo diversas consultas sobre la base de datos, lo que permitió aplicar y evaluar la funcionalidad del modelo diseñado. Cada relación presente en el diagrama Entidad-Relación fue acompañada de una descripción breve de la técnica utilizada, destacando su relevancia y aplicabilidad en el contexto del proyecto.

**DISEÑO DEL DIAGRAMA ENTIDAD-RELACIÓN**

En este apartado especificare los pasos seguidos para la realización del diagrama Entidad-Relación.

## **Captura de requisitos**

Lo primero que realizamos fue un correcto análisis de los requisitos mostrados en el enunciado para diseñar una base de datos relacional para la recuperación de datos de combustibles de España.

## **Diseño conceptual**

En segundo lugar, se capturaron todos los elementos necesarios para representar la estructura lógica de la base de datos, mostrando las entidades, sus atributos, las relaciones entre ellas y las restricciones que rigen dichas relaciones.

### **Entidades**

* Provincias → provinces.
* Municipios → municipalities.
* Localidades → localities.
* Operadores → operators.
* Combustibles → fuels.
* Precios → prices.
* Estaciones de servicio → stations.

### **Atributos**

* **provinces**:
  + *pro\_id*
  + name
* **municipalities**:
  + *mun\_id*
  + pro\_id
  + name
* **localities**:
  + *loc\_id*
  + mun\_id
  + name
* **operators**:
  + *op\_id*
  + name
* **fuels**:
  + *fuel\_id*
  + name
* **prices**:
  + *fuel\_id*
  + *st\_id*
  + name
* **stations**:
  + *st\_id*
  + loc\_id
  + op\_id
  + fuel\_id
  + cp
  + latitude
  + longitude
  + margin
  + address
  + price\_date
  + schedule
  + type

### **Relaciones**

* Gestiona → gestiona
* Tiene → tiene
* Pertenece → pertenece

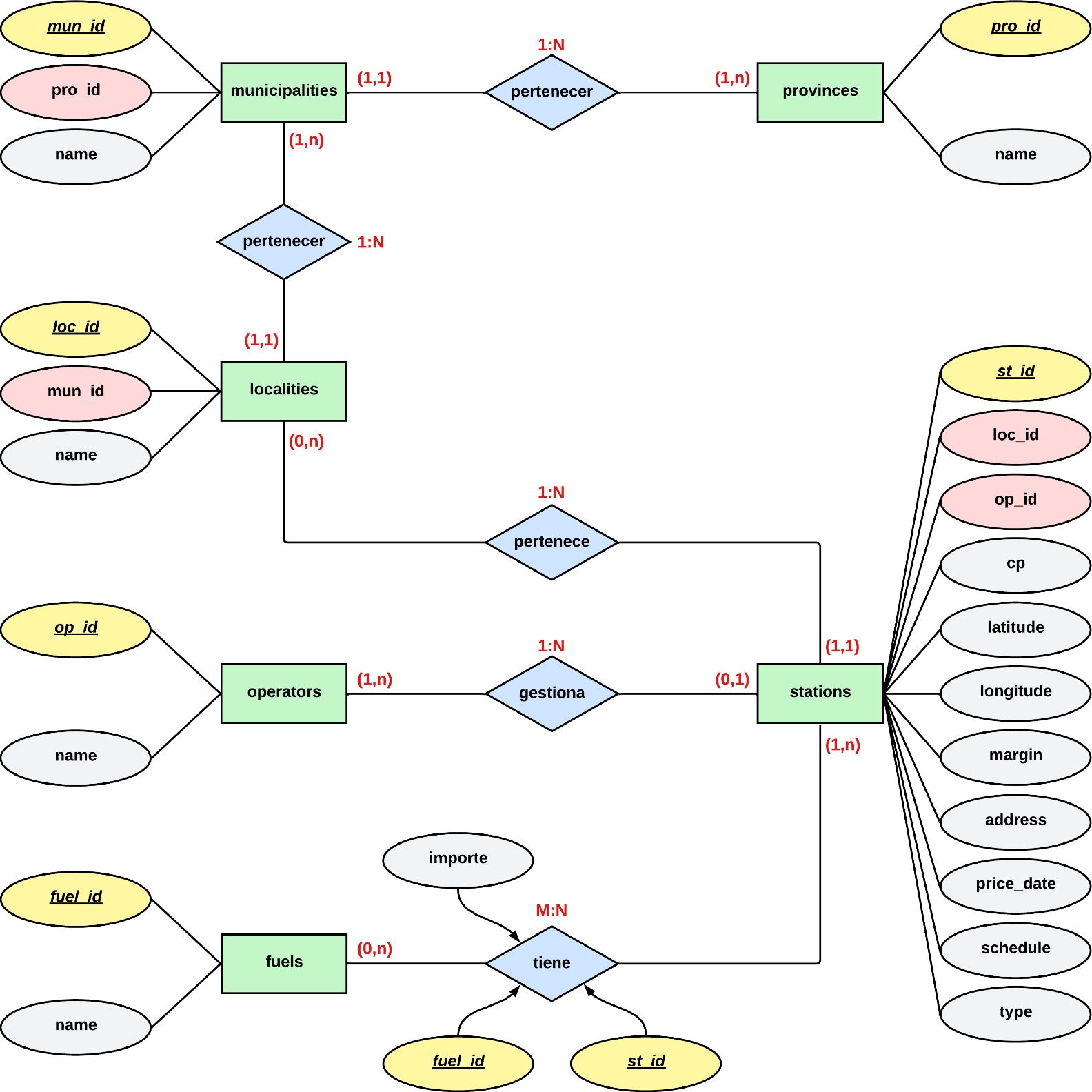
### **Relaciones entre tablas**

* municipalities → pertenece → provinces
* localities → pertenece → municipalities
* stations → pertenece → localities
* operators → gestiona → stations
* stations → tiene → fuels
* fuels → tiene → prices

### **Claves primarias y foráneas**

* **provinces**:
  + Clave primaria: *pro\_id*
* **municipalities**:
  + Clave primaria: *mun\_id*
  + Clave foránea: pro\_id
* **localities**:
  + Clave primaria: *loc\_id*
  + Clave foránea: mun\_id
* **operators**:
  + Clave primaria: *op\_id*
* **fuels**:
  + Clave primaria: *fuel\_id*
* **prices**:
  + Claves primarias: *fuel\_id, st\_id*
* **stations**:
  + Clave primaria: *st\_id*
  + Claves foráneas: loc\_id, op\_id, fuel\_id

## **Diagrama Entidad-Relación**



**Explicación de las relaciones**

A continuación, se realizará una breve explicación de las relaciones entre las entidades descritas en el diagrama entidad-relación.

### **Relación "pertenece" entre "municipalities" y "provinces"**

En esta relación, cada municipio pertenece exactamente a una provincia, y a cada provincia le pueden pertenecer varios municipios, lo que implica una relación **1:N**.

El enlace entre estas entidades se hace a través de la clave ***pro\_id***. Se generan las claves foráneas en la entidad **municipalities** que se relaciona con la clave primaria de la entidad **provinces**.

### **Relación "pertenece" entre "localities" y " municipalities"**

En esta relación, cada localidad pertenece exactamente a un municipio, y a cada municipio le pueden pertenecer varias localidades, lo que implica una relación **1:N**.

El enlace entre estas entidades se hace a través de la clave ***mun\_id***. Se generan las claves foráneas en la entidad **localities** que se relaciona con la clave primaria de la entidad **municipalities**.

### **Relación "gestiona" entre "operators" y " stations"**

En esta relación, cada operadora puede gestionar varias estaciones de servicio, y cada estación de servicio es dirigida por un único operador, lo que implica una relación **1:N**.

El enlace entre estas entidades se hace a través de la clave ***op\_id***. Se generan las claves foráneas en la entidad **stations** que se relaciona con la clave primaria de la entidad **operators**.

### **Relación "tiene" entre "stations" y " fuels"**

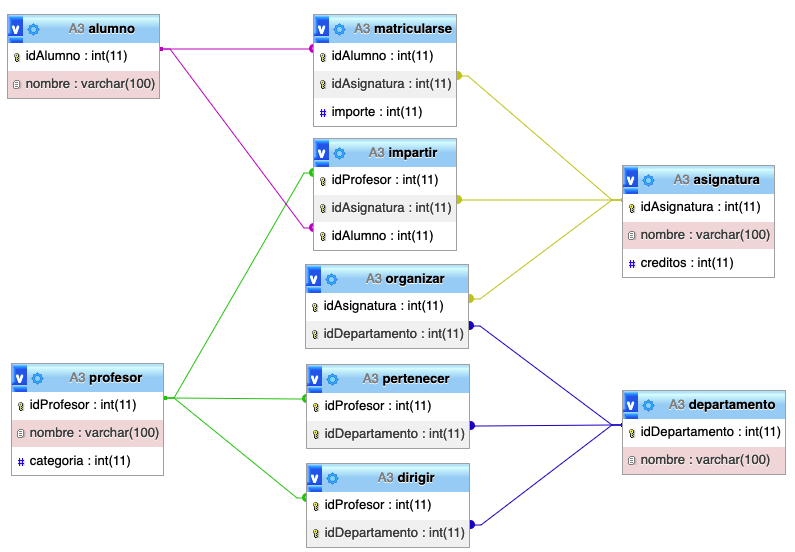
En esta relación, cada estación de servicio puede tener varios carburantes, y cada carburante puede pertenecer a varias estaciones de servicio, lo que implica una relación **M:N**.

Además, los carburantes tienen precios diferentes dependiendo de la estación que los suministre, los cuales estarían dentro de la relación “tiene”, evitando así tener duplicados en la tabla fuels. Para contemplar esto, se decide crear una tabla que represente la relación “tiene” a la que he llamado ***prices***, y que contiene el atributo precio. Además, esta tabla servirá de relación entre las entidades ***stations*** y ***fuels***.

El enlace entre estas entidades se hace a través de las claves ***st\_id*** y ***fuel\_id***. Se generan las claves foráneas en la tabla ***prices*** que se relaciona con la clave primaria de la entidad ***fuels*** y con la clave primaria de la entidad ***stations***.

## **Diseño lógico**

En este apartado muestro la transformación del diagrama entidad-relación en un esquema lógico, en el que se representa las tablas, la estructuración de datos y el modelado de restricciones disponibles.



**CONSULTAS**

### **1. Nombre de la empresa con más estaciones de servicio terrestres**

SELECT max (op\_id), operators.name as empresa

FROM operators, stations

WHERE opertaros.op\_id = stations.op\_id AND

operators.type = `T`

### **2. Nombre de la empresa con más estaciones de servicio marítimas**

SELECT max (op\_id), operators.name as empresa

FROM operators, stations

WHERE opertaros.op\_id = stations.op\_id AND

operators.type = `M`

### **3. Localización, nombre de empresa y margen de la estación con el precio más bajo para el combustible «Gasolina 95 E5» en la Comunidad de Madrid**

SELECT operators.name as nombre, stations.longitude as longitud, stations.latitude as latitud, stations.margin as margen

FROM stations, operators, fuels, prices

WHERE opertaros.op\_id = stations.op\_id AND

stations.st\_id = price.st\_id AND

fuels.fuel\_id = price.fuel\_id AND

fuels.name = `Gasolina 95 E5`

ORDER BY amount ASC

LIMIT 1

### **4. Localización, nombre de empresa y margen de la estación con el precio más bajo para el combustible «Gasóleo A» si resido en el centro de Albacete y no quiero desplazarme más de 10 KM**

SELECT operators.name as nombre, stations.longitude as longitud, stations.latitude as latitud, stations.margin as margen,

FROM stations, operators, fuels, prices

WHERE opertaros.op\_id = stations.op\_id AND

stations.st\_id = price.st\_id AND

fuels.fuel\_id = price.fuel\_id AND

fuels.name = `Gasoleo A` AND

(6371 \* acos(cos(radians(38.9942400)) \* cos(radians(stations.latitude)) \* cos(radians(stations.longitude) - radians(-1.8564300)) + sin(radians(38.9942400)) \* sin(radians(stations.latitude)))) <= 10

ORDER BY (6371 \* acos(cos(radians(38.9942400)) \* cos(radians(stations.latitude)) \* cos(radians(stations.longitude) - radians(-1.8564300)) + sin(radians(38.9942400)) \* sin(radians(stations.latitude)))) ASC

SELECT nombre, lat, lon, ( 6371 \* acos(cos(radians(41.671958)) \* cos(radians(lat)) \* cos(radians(lon) - radians(-3.685049)) + sin(radians(41.671958)) \* sin(radians(lat)))) AS distance FROM mitabla HAVING distance < 70 ORDER BY distance;

### **5. Provincia en la que se encuentre la estación de servicio marítima con el combustible «Gasolina 95 E5» más caro**

SELECT provinces.name as provincia

FROM provinces, municipalities, localities, stations, fuels, prices

WHERE provinces.pro\_id = municipalities.pro\_id AND

municipalities.mun\_id = localities.mun\_id AND

localities.loc\_id = stations.loc\_id AND

stations.st\_id = price.st\_id AND

fuels.fuel\_id = price.fuel\_id AND

fuels.name = `Gasolina 95 E5`

ORDER BY amount DESC

LIMIT 1

**CONCLUSIONES**

El diseño de una base de datos, desde la creación del diagrama entidad-relación hasta la elección de las tablas adecuadas, puede ser un proceso desafiante debido a varias dificultades.

En primer lugar, comprender completamente los requisitos del sistema y las necesidades de los usuarios es esencial para diseñar una base de datos efectiva. Si los requisitos no están claros o son contradictorios, puede ser difícil identificar las entidades y las relaciones entre ellas.

Además, la complejidad de los datos puede dificultar el modelado en el diagrama entidad-relación, aunque en nuestro caso, esto no ha sido un problema al tratarse de datos sencillos.

La normalización también es un desafío en el diseño de la base de datos, eliminar la redundancia de datos y garantizar la integridad de la base de datos es esencial. Sin embargo, aplicar correctamente las reglas de normalización puede ser complicado, especialmente al tratar con relaciones complejas o tablas con muchos atributos.

El diseño de relaciones entre las entidades es otro aspecto difícil. Determinar el tipo de relación (uno a uno, uno a muchos, muchos a muchos) y su cardinalidad puede ser un desafío, ya que implica comprender en profundidad las interacciones y dependencias entre las entidades.

Además, es importante anticiparse a los cambios futuros en los requisitos del sistema. El diseño de la base de datos debe ser lo suficientemente flexible y adaptable para acomodar los cambios sin tener que rediseñar todo el sistema.

Finalmente, nos damos cuenta de que el diseño de una base de datos desde el diagrama entidad-relación hasta la elección de las tablas correctas implica superar varias dificultades y requiere de experiencia y de un enfoque meticuloso para garantizar un diseño eficiente.