



Proyecto 2

2º semestre 2020 - Profesores J. Reutter - D. Vrgoc

Gabriel Aguirre - Matías Duhalde

1. Diagrama E/R

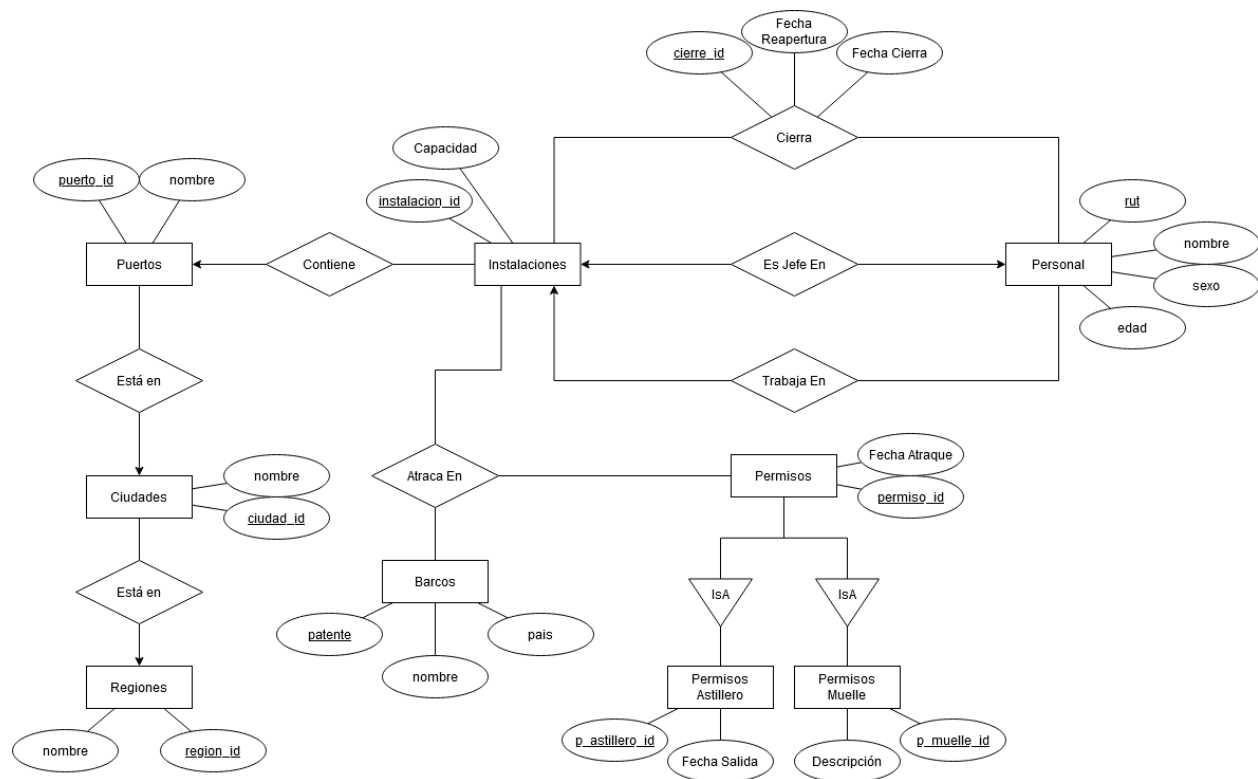


Figura 1: Diagrama Entidad-Relación

2. Esquema Relacional

regiones(region_id: int, nombre: string)

ciudades(ciudad_id: int, nombre: string)

region_ciudad(region_id: int, ciudad_id: int)

puertos(puerto_id: int, nombre: string)

ciudad_puerto(ciudad_id: int, puerto_id: int)

instalaciones(instalacion_id: int, tipo: string, capacidad: int)

puerto_instalacion(puerto_id: int FOREIGN KEY, instalacion_id: int FOREIGN KEY)

personal(rut: string, nombre: string, edad: int, sexo: string)

jefe_instalacion(instalacion_id: int FOREIGN KEY, rut: string FOREIGN KEY)

trabaja_en(rut: string FOREIGN KEY, instalacion_id: int FOREIGN KEY)

cierres(cierre_id: int, instalacion_id: int FOREIGN KEY, rut: string FOREIGN KEY, fecha_cierre: timestamp, fecha_reapertura: timestamp)

barcos(patente: string, pais: string, nombre: string)

permisos(permiso_id: int, fecha_ataque: timestamp)

permisos_astillero(p_astillero_id: int, permiso_id: int FOREIGN KEY, fecha_salida: timestamp)

permisos_muelle(p_muelle_id: int, permiso_id: int FOREIGN KEY, descripcion: string)

atraca_en(permiso_id: int FOREIGN KEY, instalacion_id: int FOREIGN KEY, patente: string FOREIGN KEY)

3. Justificación

A continuación, se analizará cada tabla y sus dependencias funcionales, para mostrar que el esquema se encuentra normalizado en **BCNF**.

3.1. Tabla regiones

Atributos:

- region_id (PRIMARY KEY)
- nombre

Dependencias funcionales:

- region_id \rightarrow nombre

3.2. Tabla ciudades

Atributos:

- ciudad_id (PRIMARY KEY)
- nombre

Dependencias funcionales:

- ciudad_id \rightarrow nombre

3.3. Tabla region_ciudad

Atributos:

- region_id (PRIMARY KEY, FOREIGN KEY)
- ciudad_id (PRIMARY KEY, FOREIGN KEY)

Dependencias funcionales: Esta tabla no posee dependencias funcionales.

3.4. Tabla puertos

Atributos:

- puerto_id (PRIMARY KEY)
- nombre

Dependencias funcionales:

- puerto_id \rightarrow nombre

3.5. Tabla ciudad_puerto

Atributos:

- ciudad_id (PRIMARY KEY, FOREIGN KEY)
- puerto_id (PRIMARY KEY, FOREIGN KEY)

Dependencias funcionales: Esta tabla no posee dependencias funcionales.

3.6. Tabla instalaciones

Atributos:

- instalacion_id (PRIMARY KEY)
- tipo
- capacidad

Dependencias funcionales:

- instalacion_id \rightarrow tipo, capacidad

3.7. Tabla puerto_instalacion

Atributos:

- puerto_id (PRIMARY KEY, FOREIGN KEY)
- instalacion_id (PRIMARY KEY, FOREIGN KEY)

Dependencias funcionales: Esta tabla no posee dependencias funcionales.

3.8. Tabla personal

Atributos:

- rut (PRIMARY KEY)
- nombre
- edad
- sexo

Dependencias funcionales:

- rut \rightarrow nombre, edad, sexo

3.9. Tabla jefe_instalacion

Atributos:

- instalacion_id (PRIMARY KEY, FOREIGN KEY)
- rut (FOREIGN KEY)

Dependencias funcionales:

- instalacion_id \rightarrow rut

3.10. Tabla trabaja_en

Atributos:

- rut (PRIMARY KEY, FOREIGN KEY)
- instalacion_id (FOREIGN KEY)

Dependencias funcionales:

- rut \rightarrow instalacion_id

3.11. Tabla cierres

Atributos:

- cierre_id (PRIMARY KEY)
- instalacion_id (FOREIGN KEY)
- rut (FOREIGN KEY)
- fecha_cierre
- fecha_reapertura

Dependencias funcionales:

- cierre_id \rightarrow instalacion_id, rut, fecha_cierre, fecha_reapertura

3.12. Tabla barcos

Atributos:

- patente (PRIMARY KEY)
- pais
- nombre

Dependencias funcionales:

- patente \rightarrow pais, nombre

3.13. Tabla permisos

Atributos:

- permiso_id (PRIMARY KEY)
- fecha_atraque

Dependencias funcionales:

- permiso_id \rightarrow fecha_atraque

3.14. Tabla permisos_astillero

Atributos:

- p_astillero_id (PRIMARY KEY)
- permiso_id (FOREIGN KEY)
- fecha_salida

Dependencias funcionales:

- p_astillero_id \rightarrow permiso_id, fecha_salida

3.15. Tabla permisos_muelle

Atributos:

- p_muelle_id (PRIMARY KEY)
- permiso_id (FOREIGN KEY)
- descripcion

Dependencias funcionales:

- p_muelle_id \rightarrow permiso_id, descripcion

3.16. Tabla atraca_en

Atributos:

- permiso_id (PRIMARY KEY)
- instalacion_id (FOREIGN KEY)
- patente (FOREIGN KEY)

Dependencias funcionales:

- $\text{permiso_id} \rightarrow \text{instalacion_id}, \text{patente}$

Se puede ver que para todas las tablas, existe a lo más una relación de dependencia funcional. Además, para toda dependencia $X \rightarrow Y$, se tiene que $Y \not\subseteq X$ (es decir, no son triviales), y X es llave de la tabla. Por lo tanto, el esquema estaría justificado según **BCNF**.

4. Consultas

4.1. Consulta 1

■ Muestre todos los *puertos* junto la ciudad a la que son asignados.

```
SELECT ciudades.ciudad_id, ciudades.nombre, puertos.puerto_id,
puertos.nombre FROM puertos
INNER JOIN ciudad_puerto
ON ciudad_puerto.puerto_id = puertos.puerto_id
INNER JOIN ciudades
ON ciudad_puerto.ciudad_id = ciudades.ciudad_id;
```

4.2. Consulta 2

■ Muestre todos los jefes de las instalaciones del puerto con nombre 'Mejillones'.

```
SELECT personal.rut, personal.nombre, edad, sexo,
jefe_instalacion.instalacion_id FROM personal
INNER JOIN jefe_instalacion
ON personal.rut = jefe_instalacion.rut
INNER JOIN puerto_instalacion
ON jefe_instalacion.instalacion_id = puerto_instalacion.instalacion_id
INNER JOIN puertos
ON puerto_instalacion.puerto_id = puertos.puerto_id
WHERE puertos.nombre ILIKE '%Mejillones%';
```

4.3. Consulta 3

■ Muestre todos los puertos que tienen al menos un astillero.

```
SELECT DISTINCT puertos.puerto_id, puertos.nombre FROM puertos
INNER JOIN puerto_instalacion
ON puertos.puerto_id = puerto_instalacion.puerto_id
INNER JOIN instalaciones
ON instalaciones.instalacion_id = puerto_instalacion.instalacion_id
WHERE instalaciones.tipo ILIKE '%astillero%';
```

4.4. Consulta 4

■ Muestre todas las veces que el barco 'Calypso' ha atracado en 'Arica'.

```
SELECT permisos.permiso_id, atraca_en.instalacion_id, instalaciones.tipo,
permisos.fecha_atraque FROM permisos
INNER JOIN atraca_en
ON atraca_en.permiso_id = permisos.permiso_id
INNER JOIN barcos
ON barcos.patente = atraca_en.patente
INNER JOIN instalaciones
```



```
ON instalaciones.instalacion_id = atraca_en.instalacion_id
INNER JOIN puerto_instalacion
ON instalaciones.instalacion_id = puerto_instalacion.instalacion_id
INNER JOIN puertos
ON puertos.puerto_id = puerto_instalacion.puerto_id
INNER JOIN ciudad_puerto
ON puertos.puerto_id = ciudad_puerto.puerto_id
INNER JOIN ciudades
ON ciudad_puerto.ciudad_id = ciudades.ciudad_id
WHERE barcos.nombre ILIKE '%Calypso%' AND ciudades.nombre ILIKE '%Arica%';
```

4.5. Consulta 5

■ Muestre la edad promedio de los trabajadores de cada puerto.

```
SELECT puertos.puerto_id, puertos.nombre, AVG(personal.edad) AS promedio_edad FROM per
INNER JOIN trabaja_en
ON personal.rut = trabaja_en.rut
INNER JOIN puerto_instalacion
ON puerto_instalacion.instalacion_id = trabaja_en.instalacion_id
INNER JOIN puertos
ON puertos.puerto_id = puerto_instalacion.puerto_id
GROUP BY puertos.puerto_id, puertos.nombre;
```

4.6. Consulta 6

■ Muestre el puerto que ha recibido más barcos en Agosto del 2020.

```
SELECT t.nombre, barcos_atracados
FROM (SELECT puertos.nombre, COUNT(DISTINCT atraca_en.patente)
      AS barcos_atracados FROM puertos
      INNER JOIN puerto_instalacion
      ON puertos.puerto_id = puerto_instalacion.puerto_id
      INNER JOIN atraca_en
      ON puerto_instalacion.instalacion_id = atraca_en.instalacion_id
      INNER JOIN permisos
      ON atraca_en.permiso_id = permisos.permiso_id
      WHERE DATE_PART('month', permisos.fecha_atraque) = 8 AND
      DATE_PART('year', permisos.fecha_atraque) = 2020
      GROUP BY puertos.nombre) AS t
ORDER BY barcos_atracados DESC
LIMIT 1;
```

5. Consideraciones y detalles de la entrega

1. Para que sea más fácil entender el diagrama (sea más parecido al lenguaje natural), en el caso de las relaciones, algunas tienen un nombre distinto al del esquema. Los casos particulares donde esto sucede son:
 - a)* **Está en** (ciudades-puertos) → `ciudad_puerto`
 - b)* **Está en** (regiones-ciudades) → `region_ciudad`
 - c)* **Contiene** (puertos-instalaciones) → `puerto_instalacion`
 - d)* **Es Jefe En** (instalaciones-personal) → `jefe_instalacion`
 - e)* **Trabaja En** (instalaciones-personal) → `trabaja_en`
 - f)* **Cierra** (instalaciones-personal) → `cierres`
 - g)* **Atraca En** (permisos-instalaciones-barcos) → `atraca_en`
2. Según lo discutido en la [issue 132](#), en la consulta 4, con *Arica*, el enunciado se refiere al nombre de la ciudad (no el nombre del puerto ni de la región).
3. Según lo discutido en la [issue 127](#), en la consuta 6, si un barco atraca más de una vez en el mismo puerto en el mismo periodo de tiempo, este sólo cuenta una vez (notar keyword DISTINCT).