

# Hoteles rurales

**Proyecto completo:**

**Entidad-Relación, paso a tablas e implementación en SQL**

Isabel Riomoros



# Especificación de requisitos

# 1 Hoteles rurales

La Comunidad de Madrid desea guardar información sobre los Hoteles rurales que existen en la Comunidad, dispone de la siguiente información:

- ☐ Un **Hotel rural** se identifica por un nombre, tiene una dirección, teléfonos de contacto y una **persona de contacto** que pertenece al personal del Hotel. También le interesa la localidad o pueblo de la Comunidad en el que está ubicado, con el fin de saber los municipios que tienen necesidad de ser subvencionados para la futura realización de Hoteles y la cantidad de Hoteles por municipio.
- ☐ En cada Hotel trabajan una **serie de personas** que se identifican por un código personal. Se requiere conocer el nombre completo(nombre, apellido1 y apellido2), la dirección y el NIF.
- ☐ En un Hotel pueden trabajar varias personas, pero **una persona sólo puede trabajar en un Hotel**.
- ☐ Los Hoteles se alquilan por **habitaciones**, se desea conocer cuántas habitaciones componen el Hotel de qué tipo (individuales, dobles, triples) es cada una de las habitaciones, si poseen cuarto de baño y precio.
- ☐ En algunos de estos Hoteles se realizan **actividades** multiaventura organizadas para huéspedes (senderismo, bicicleta, etc.)
- ☐ Esta actividad se identifica por un código. También es interesante saber el nombre de la actividad, la descripción y el nivel de dificultad de dicha actividad.
- ☐ Las actividades se realizan **un día a la semana**, pero puede haber algún día en el que no se practique ninguna actividad.

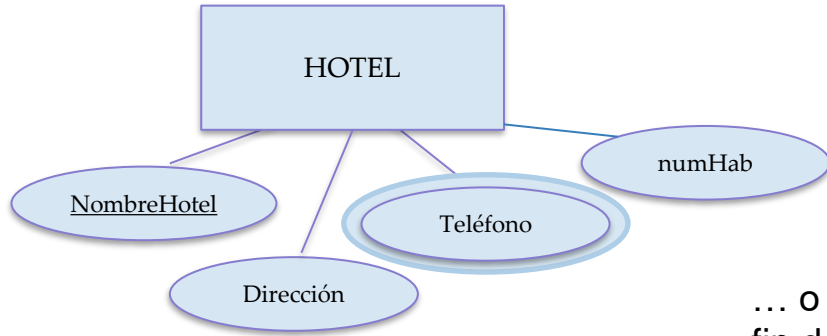


# Diseño conceptual

## Modelo entidad-relación

- ❑ **Entidad**, cosa u objeto en el mundo real que es distinguible de todos los demás objetos . **“Una cosa que se puede identificar claramente”**
- ❑ **Un atributo** describe las propiedades que posee cada entidad
- ❑ **Relación**, *asociación entre diferentes entidades*. **“Vinculación entre entidades”**

# Identificación de las entidades y atributos

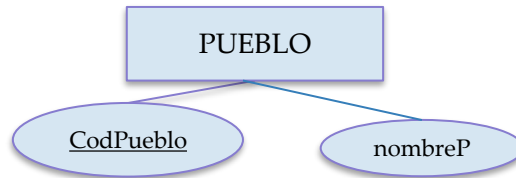


Un **Hotel rural** se identifica por un nombre, tiene una dirección, teléfonos de contacto, el número de habitaciones y la población en la que está ubicado.

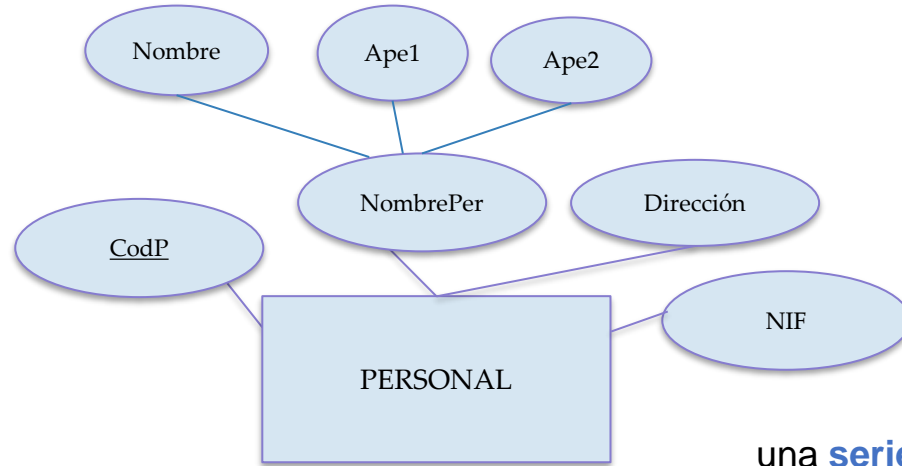
Como puede tener varios teléfonos, el atributo teléfono es multivalorado.

... o pueblo de la Comunidad en el que está ubicado, con el fin de saber los municipios que tienen necesidad de ser subvencionados para la futura realización de Hoteles y la cantidad de Hoteles por municipio.

Nos interesa tener una tabla pueblos, ya que los vamos a necesitar para hacer consultas, además así estarán guardados sólo una vez, la entidad pueblo tendrá el código identificativo y el nombre del pueblo



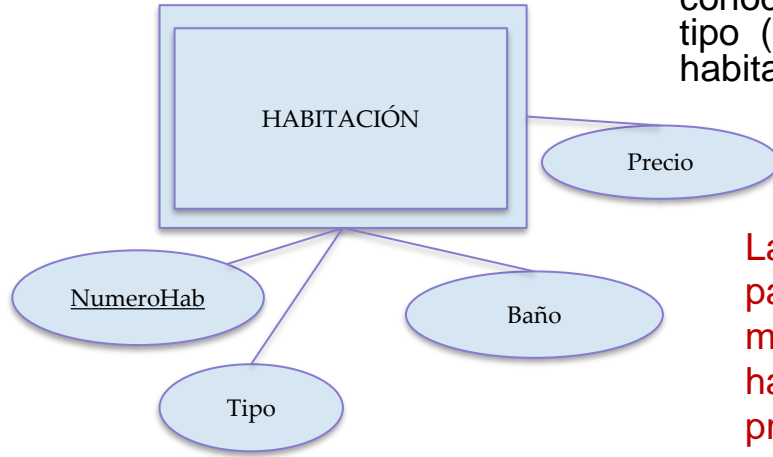
# Identificación de las entidades y atributos



una **serie de personas** que se identifican por un código personal. Se requiere conocer el nombre completo, la dirección y el NIF. **El atributo nombre es compuesto, porque está formado por el nombre, primer apellido y segundo apellido.**

# Identificación de las entidades

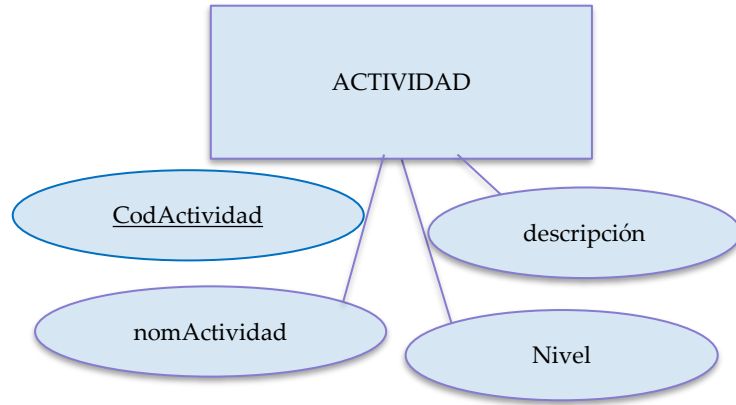
- Los Hoteles se alquilan por **habitaciones**, se desea conocer cuántas habitaciones componen el Hotel de qué tipo (individuales, dobles, triples) es cada una de las habitaciones, si poseen cuarto de baño y precio.



La **entidad habitación es débil**, ya que no tiene un atributo para ser clave primaria. El número de habitación puede ser el mismo en dos hoteles (105,106,.. ,203,...) por lo que habitación no tiene atributos suficientes para formar clave primaria. Su existencia depende de la entidad hotel.

*Recordamos que una entidad débil es una entidad cuyas propiedades o atributos no la identifican completamente, sino que sólo la identifican de forma parcial. Tiene que participar en una relación que ayude a identificarse.*

## Identificación de las entidades y atributos

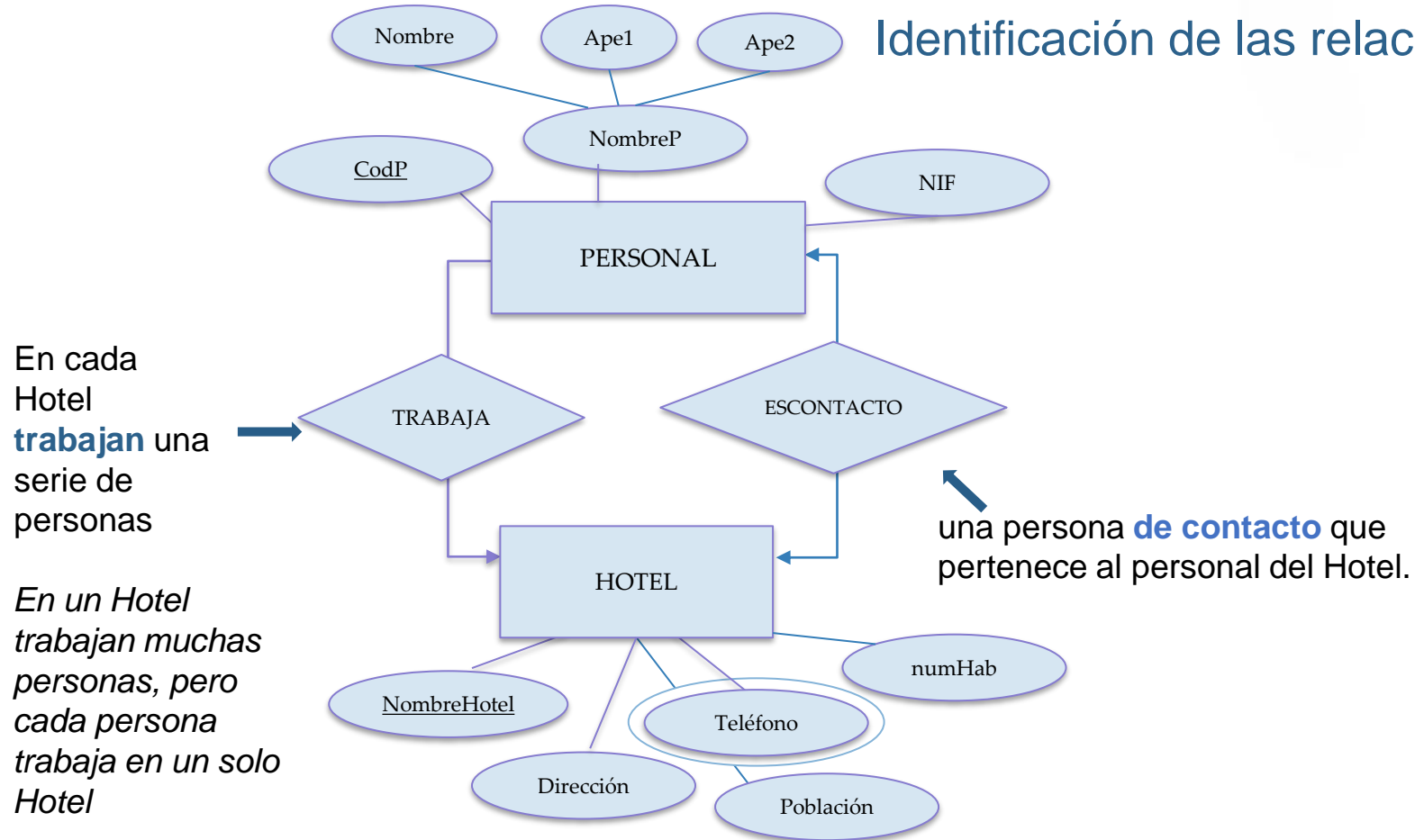


- En algunos de estos Hoteles se realizan **actividades** multiaventura organizadas para huéspedes (senderismo, bicicleta, etc.)

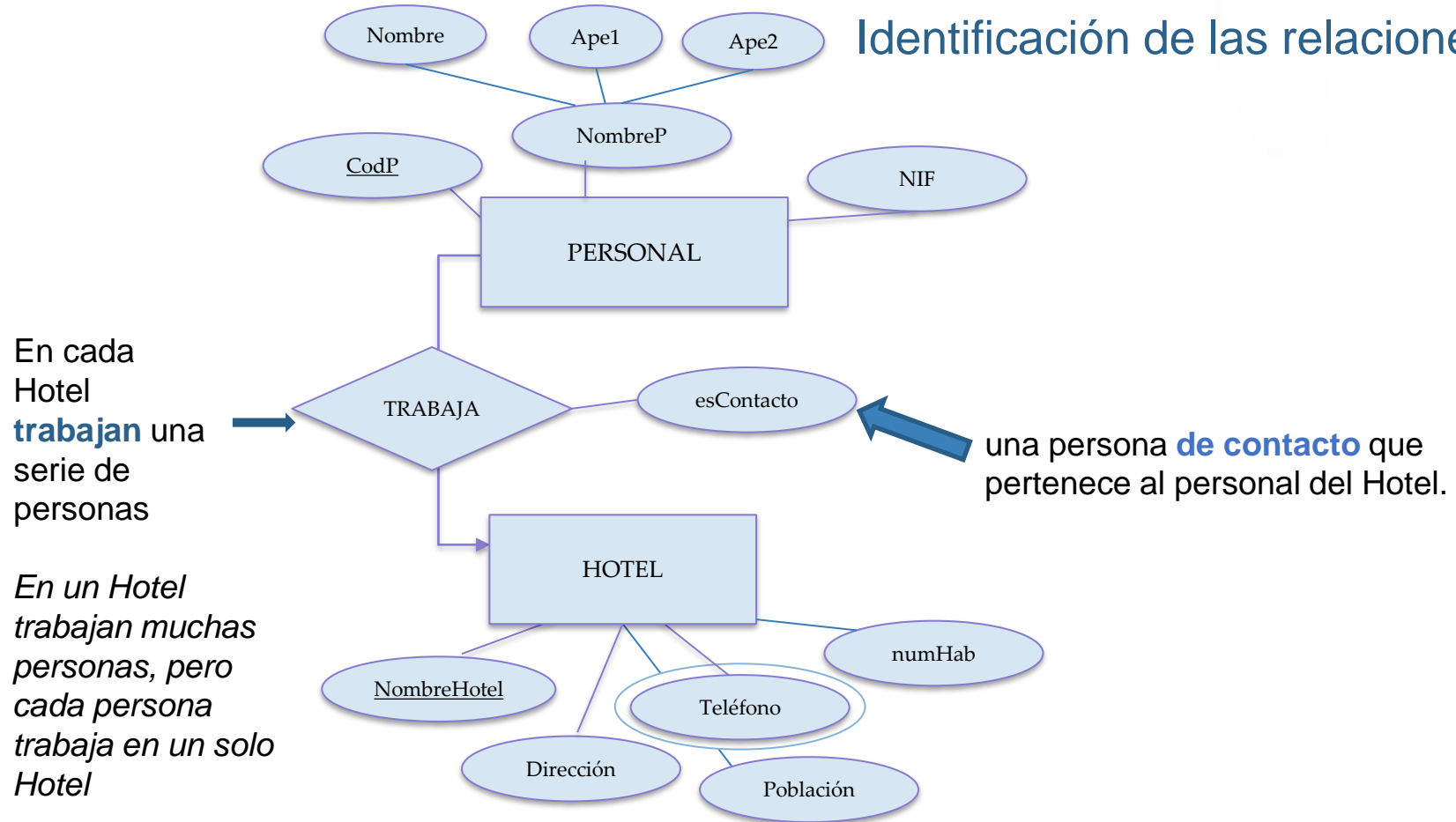
- Esta actividad se identifica por un código. También es interesante saber el nombre de la actividad, la descripción y el nivel de dificultad de dicha actividad.



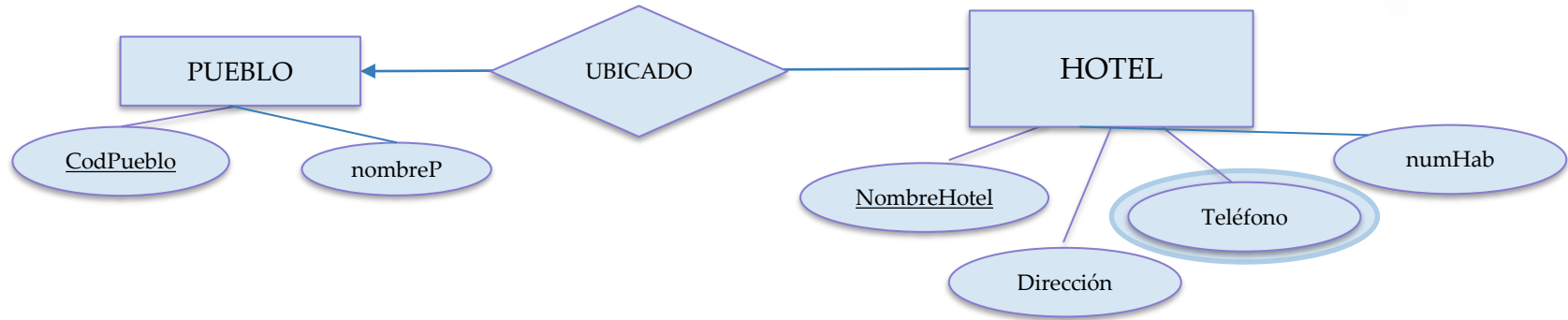
## Identificación de las relaciones



## Identificación de las relaciones

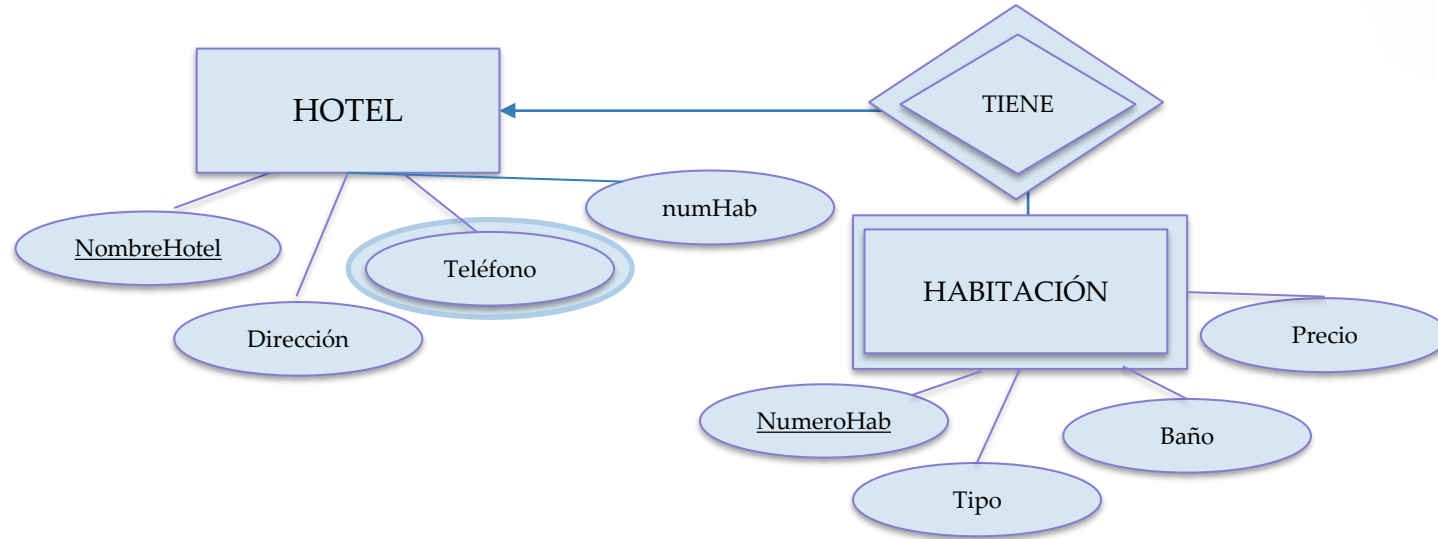


## Identificar las relaciones y sus posibles atributos



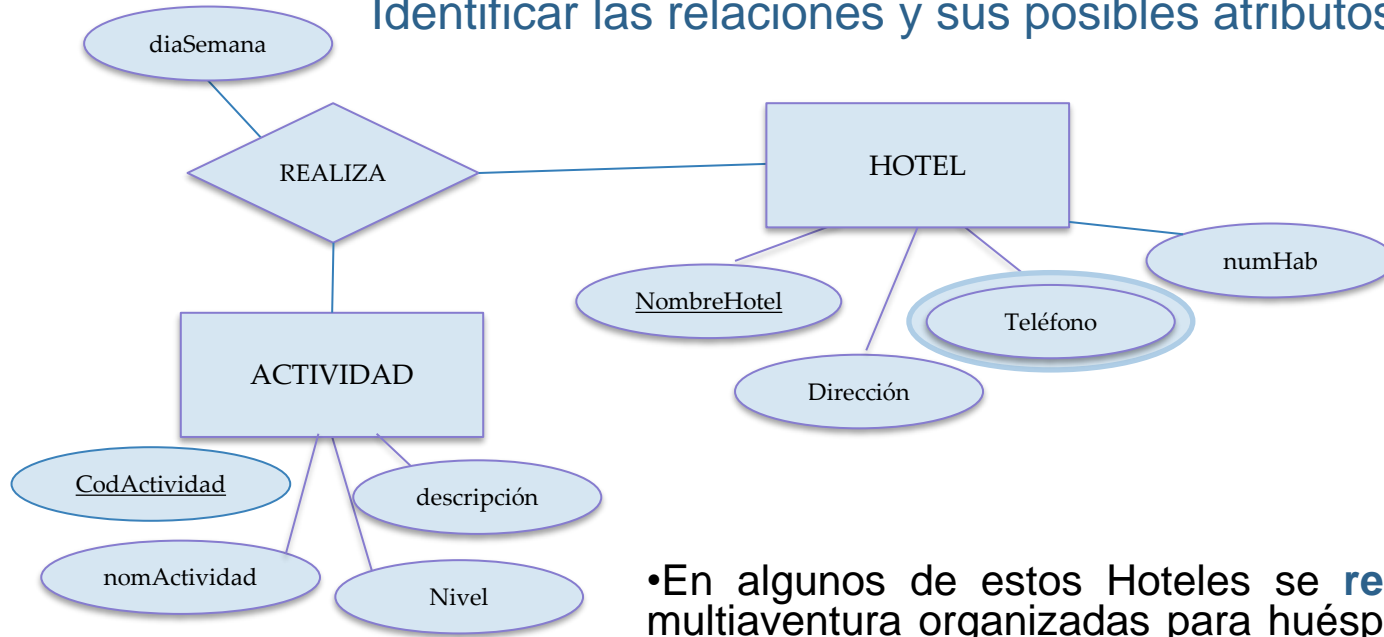
También le interesa la localidad o pueblo de la Comunidad en el que está ubicado, con el fin de saber los municipios que tienen necesidad de ser subvencionados para la futura realización de Hoteles y la cantidad de Hoteles por municipio.

## Identificar las relaciones y sus posibles atributos



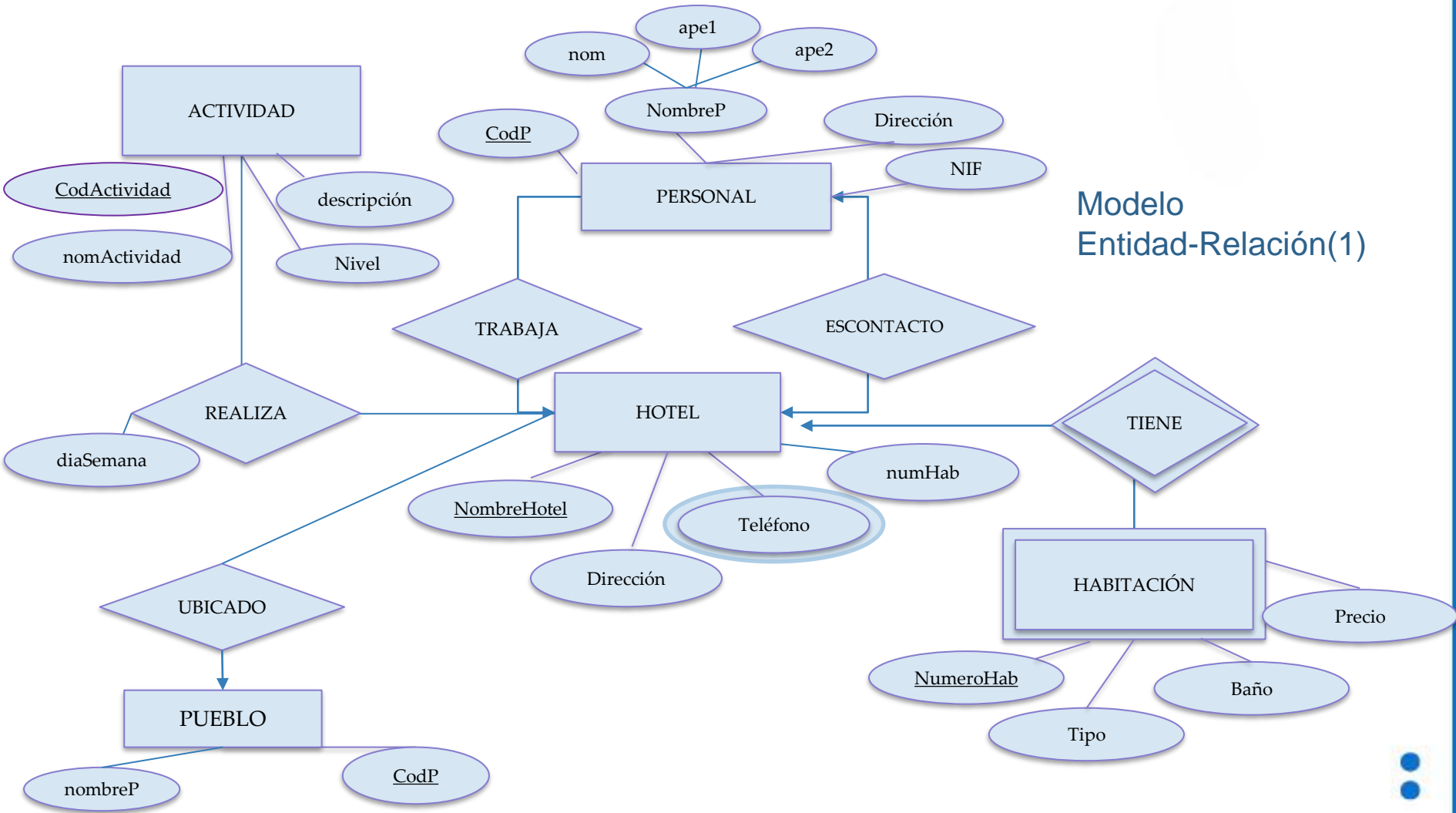
- Los Hoteles se alquilan por **habitaciones**, se desea conocer cuántas habitaciones componen el Hotel de qué tipo (individuales, dobles, triples) es cada una de las habitaciones, si poseen cuarto de baño y precio.

## Identificar las relaciones y sus posibles atributos

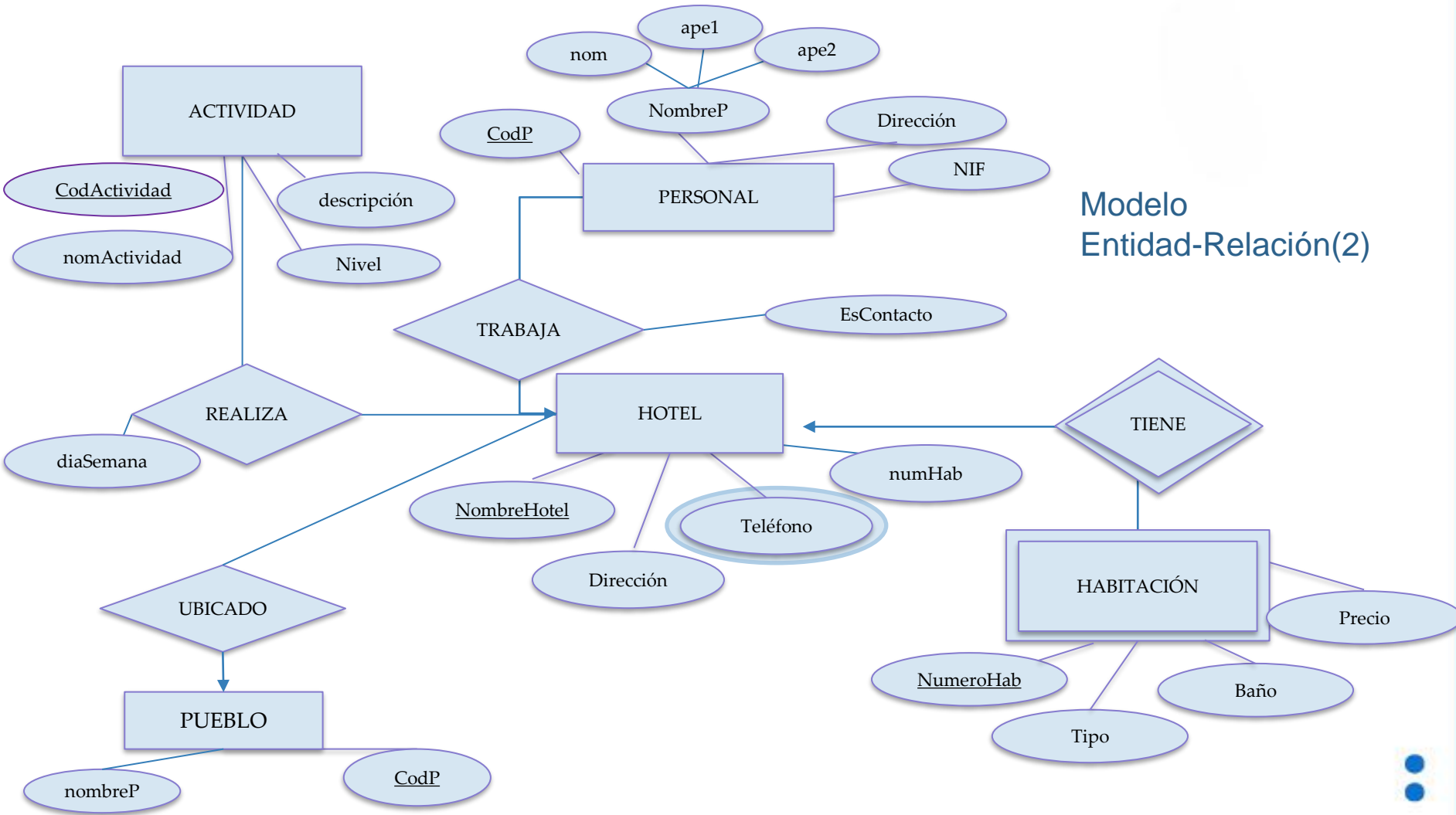


•En algunos de estos Hoteles se **realizan** actividades multiaventura organizadas para huéspedes (senderismo, bicicleta, etc.)

•Las actividades se realizan **un día a la semana**, pero puede haber algún día en el que no se practique ninguna actividad.



## Modelo Entidad-Relación(2)



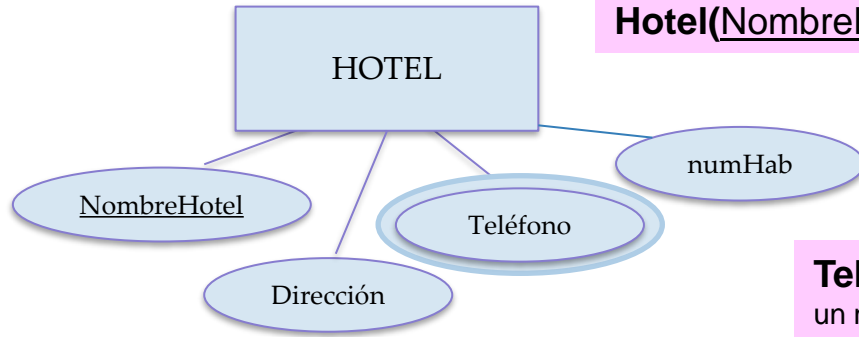
# Diseño conceptual

## Paso a tablas → Modelo relacional

- Una **base de datos relacional** es un conjunto de **tablas**, a cada una de las cuales se le asigna un nombre exclusivo.
- Cada **fila** de la **tabla** representa una colección de valores de datos relacionados entre sí, se le denomina **tupla**. Esos valores se pueden interpretar como hechos que describen una entidad o un vínculo entre entidades del mundo real.
- Todos los valores de una columna tienen el mismo tipo de datos



## Paso a tablas



**Hotel**(NombreHotel, Dirección, numHab)

Atributo multivalorado

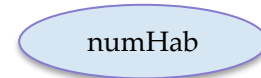


**Telefono**(Nhotel,numTel)

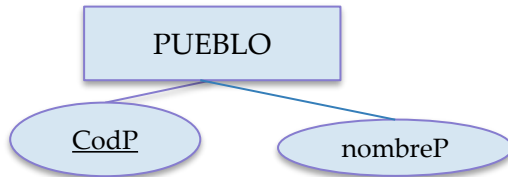
un número de teléfono **no** puede estar compartido por dos alojamientos,  
**Si** un número de teléfono pudiera estar compartido por dos hoteles la  
tabla quedaría:

**Telefono**(Nhotel,numTel)

Atributo calculado

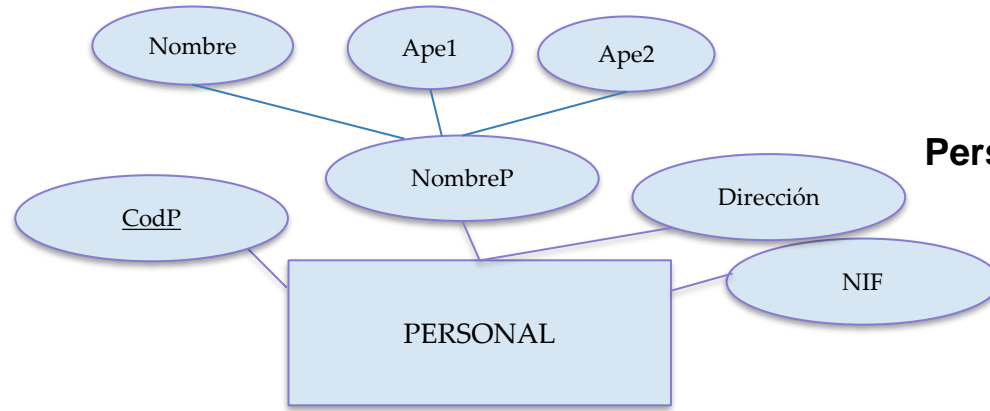


Lo vamos a calcular sumando uno cada vez que insertemos una  
habitación en un hotel determinado, con un disparador(trigger)



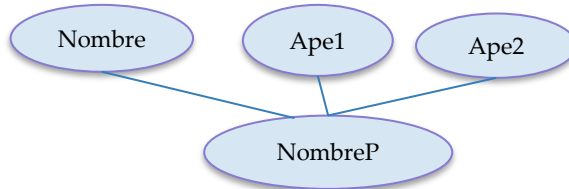
**Pueblo**(CodP, nombreP)

## Paso a tablas



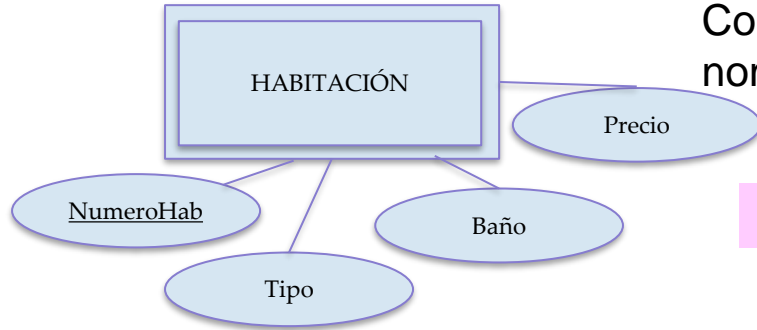
**Personal** (CodP, NombreP, Dirección, NIF)

### Atributo compuesto



**Personal** (CodP, Nombre, Ape1, Ape2, Dirección, NIF)

## Paso a tablas



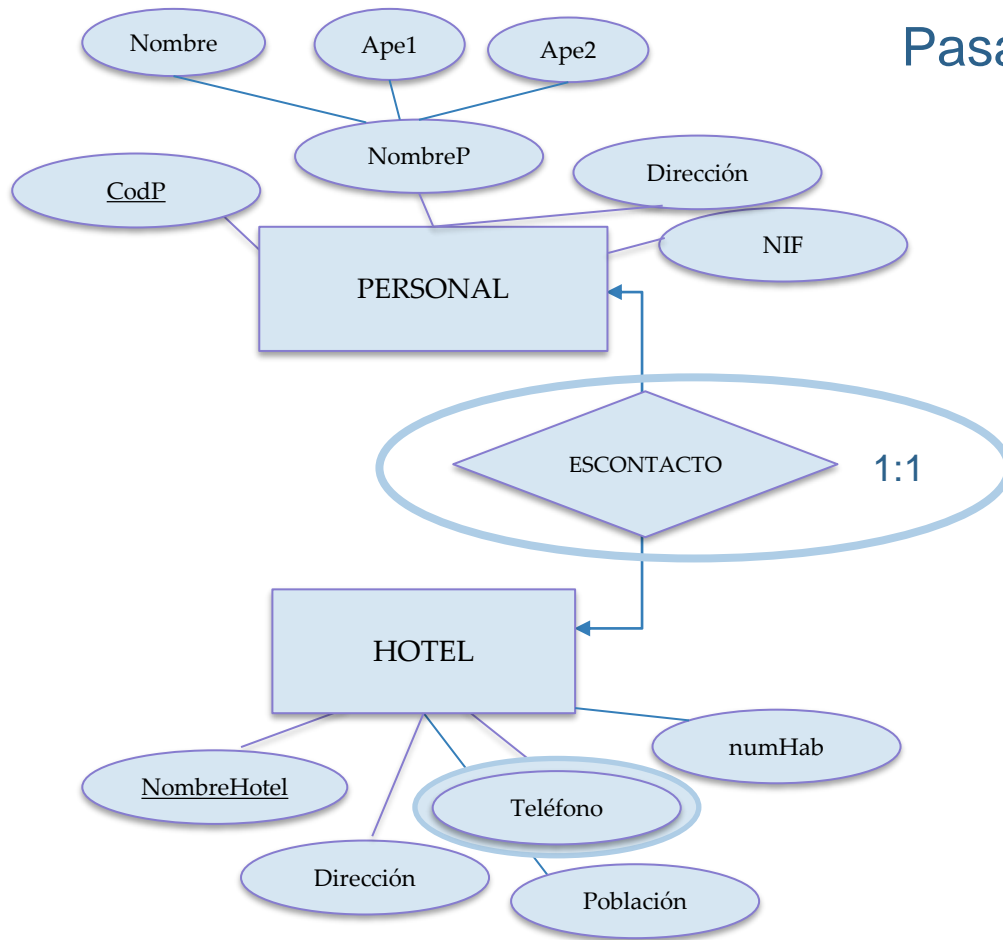
Como **Habitación** es una **entidad débil** de Hotel, el nombre del hotel formará parte de la clave primaria

**habitación**(NomHotel, NumeroHab, Tipo, Baño, Precio)



**actividad**(CódActividad, NomActividad, Descripción, Nivel)

## Pasar las relaciones a tablas



## Paso a tablas

**Persona de contacto del Hotel**, relación **esContacto**, relación 1:1, en este caso se podría poner la clave de una relación en la otra indistintamente, pero si se pone la clave primaria de Hotel en personal, ese campo será nulo muchas veces, ya que de todas las personas que trabajan en el Hotel sólo hay una que es la de contacto. Tenemos dos opciones para pasar esta relación a tabla

Una opción sería poner contacto(clave ajena que referencia a persona) en la relación Hotel.



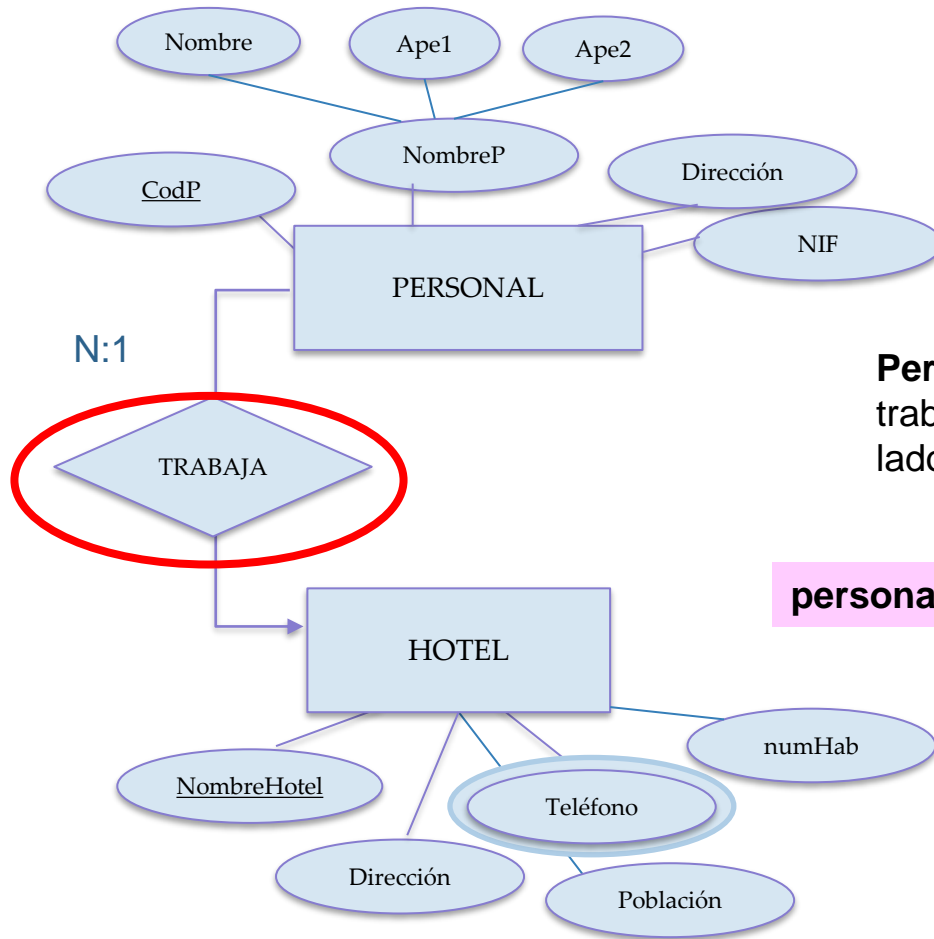
ESCONTACTO

**Hotel**(NombreHotel, Dirección,numHab,*Contacto*)

Otra opción es crear tabla con los hoteles y el contacto, la clave sólo será el hotel, porque un hotel sólo puede tener una persona de contacto.

**EsContacto**(NombreHote, *Contacto*)

# Pasar las relaciones a tablas

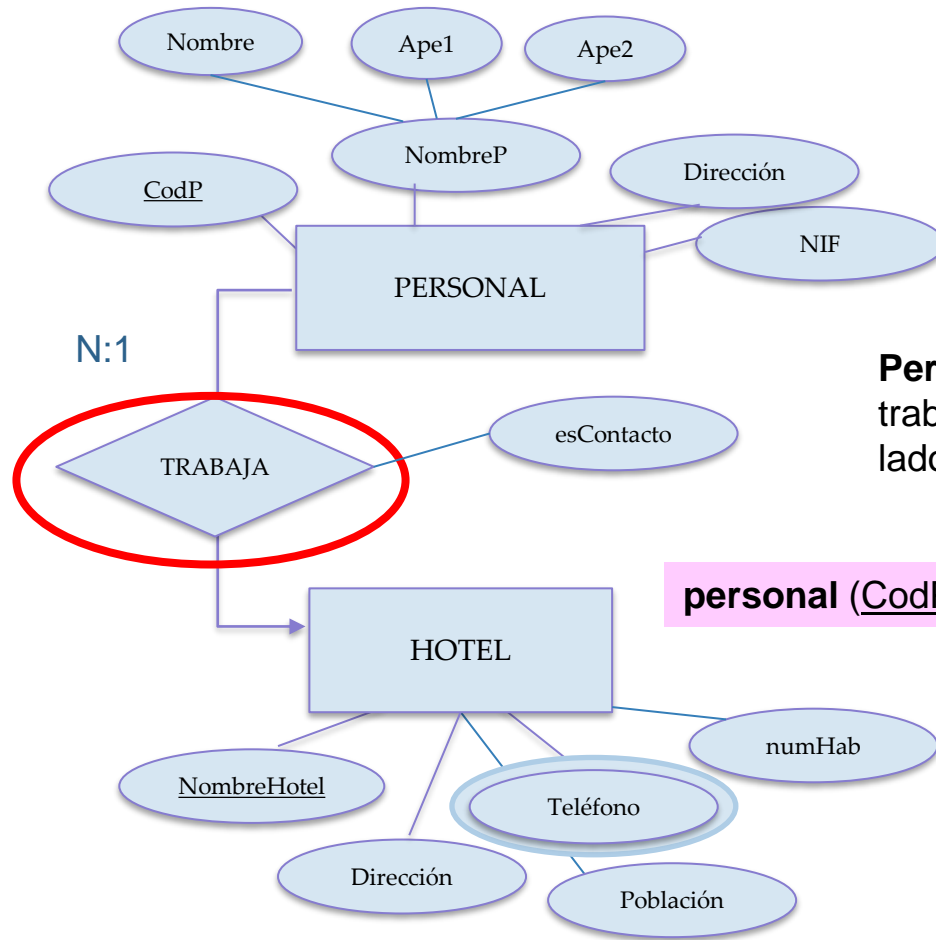


**Persona trabaja en Hotel**, relación N:1, (n personas trabajan en un Hotel) Se coloca la clave primaria del lado 1 en la entidad que está en la parte N.

**personal** (CodP, NombreP, Dirección, NIF, *NomHotel*)

Trabaja

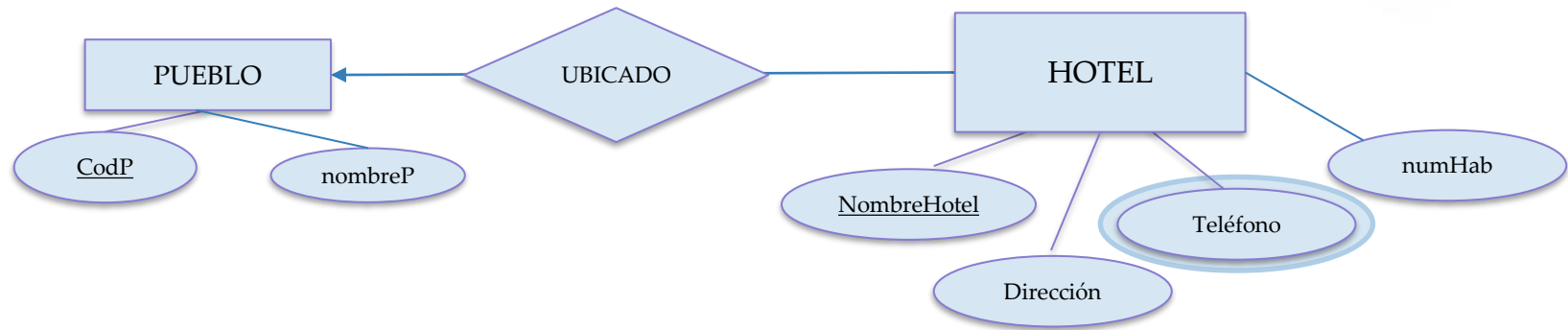
## Pasar las relaciones a tablas Modelo 2



**Persona trabaja en Hotel**, relación N:1, (n personas trabajan en un Hotel) Se coloca la clave primaria del lado 1 en la entidad que está en la parte N.

**personal** (CodP, NombreP, Dirección, NIF, NomHotel, esContacto)

Trabaja



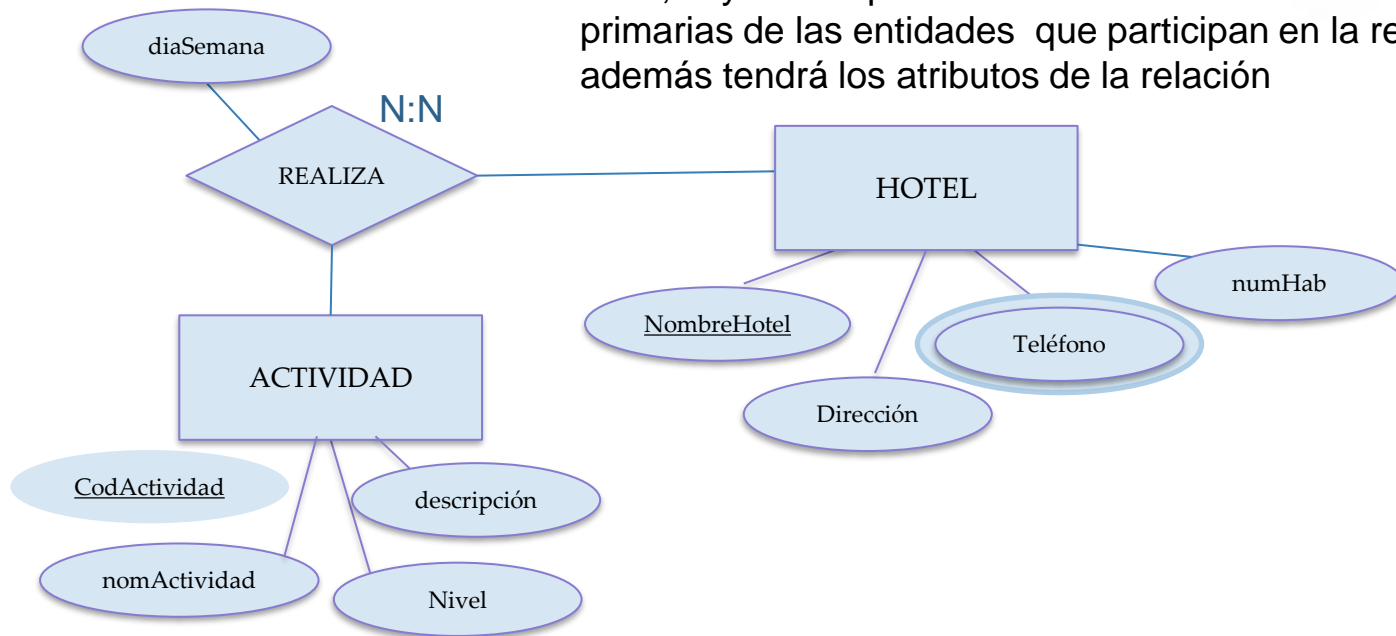
Como es una relación 1:N, se coloca la clave primaria del lado 1 en la entidad que está en la parte N. Luego la tabla hotel quedaría

**Hotel(NombreHotel, Dirección, numHab, *codigoP*)**



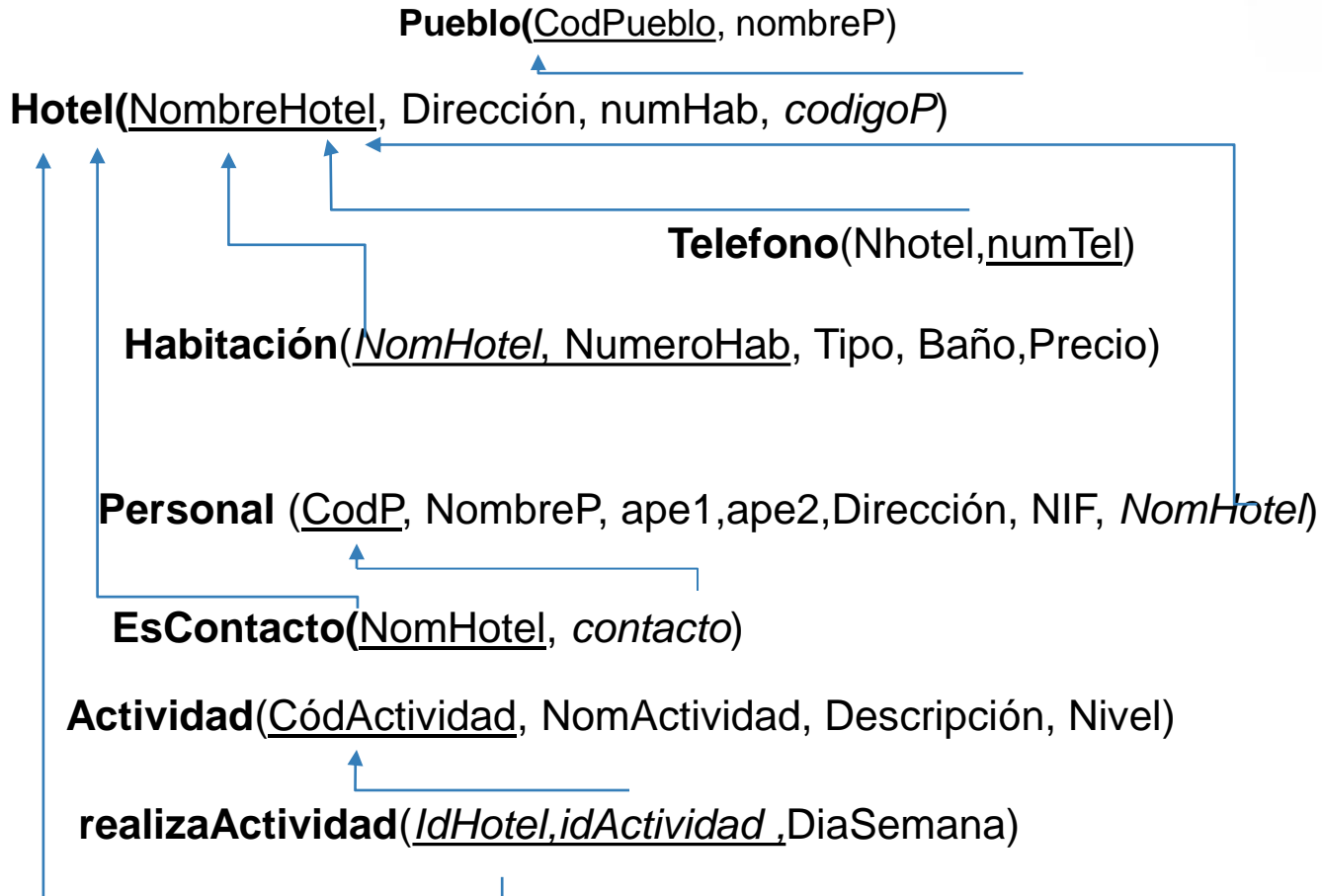
## Pasar las relaciones a tablas

Hotel realiza actividad, relación N:N se convierte en una tabla, cuya clave primaria es la unión de las claves primarias de las entidades que participan en la relación, además tendrá los atributos de la relación

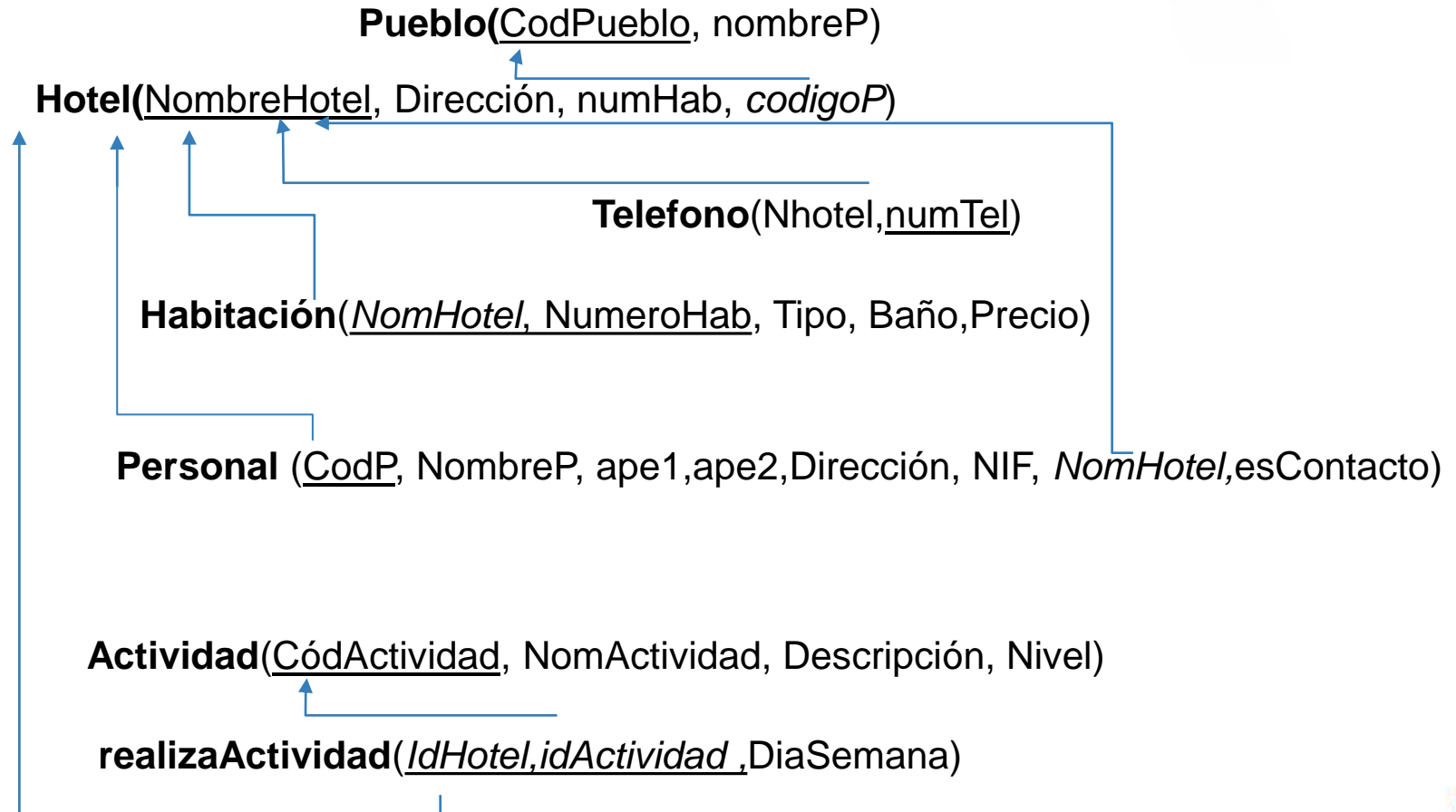


**realizaActividad(IdHotel,idActividad, DiaSemana)**

# Modelo relacional- 1



## Modelo relacional-2



# Implementación

Definiré las tablas para la opción 1, la opción 2 está en el script que os pasaré con la documentación



# Implementación SQL(1)

```
CREATE TABLE pueblo (  
  codPueblo SMALLINT PRIMARY KEY,  
  nombreP VARCHAR(30)  
);
```

```
CREATE TABLE hotel (  
  num SMALLINT PRIMARY KEY,  
  nombreHotel VARCHAR(30),  
  direccion VARCHAR(100) NOT NULL,  
  numHab SMALLINT DEFAULT 0,  
  codigoP SMALLINT,  
  FOREIGN KEY (codigoP)  
  REFERENCES pueblo(codPueblo)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE  
);
```

Al implementarlo hemos puesto un  
campo numérico para que resulte  
más fácil construir las tablas

```
CREATE TABLE telefono(  
  numTel VARCHAR(9),  
  numeroHotel smallint,  
  PRIMARY KEY (numTel),  
  FOREIGN KEY (numeroHotel) REFERENCES hotel(num)  
);
```

```
CREATE TABLE personal (  
  codP SMALLINT,  
  NIF VARCHAR(9) NOT NULL UNIQUE,  
  nombre VARCHAR(30) NOT NULL,  
  apel1 VARCHAR(30) NOT NULL,  
  ape2 VARCHAR(30) NOT NULL,  
  direccion VARCHAR(100),  
  nomHotel SMALLINT,  
  PRIMARY KEY (codP),  
  FOREIGN KEY (nomHotel) REFERENCES hotel(num)  
  ON DELETE CASCADE ON UPDATE CASCADE  
);
```

**Si se elimina un hotel, se elimina todo  
el personal que trabaja en él**

## Implementación SQL(1)

### ciclo referencial entre Hotel y PERSONAL(opción 1)

```
ALTER TABLE hotel  
ADD FOREIGN KEY (contacto)  
REFERENCES personal(codP)  
ON DELETE RESTRICT  
ON UPDATE CASCADE;
```

Si se elimina la persona de contacto de un hotel, no se eliminará el hotel

### Transformación de la relación 1:1 en tabla (opción 2)

```
CREATE TABLE esContactol (  
numeroH SMALLINT PRIMARY KEY,  
contacto SMALLINT NOT NULL UNIQUE  
FOREING KEY numeroH REFERENCES hotel(num)  
ON DELETE CASCADE ON UPDATE CASCADE ,  
FOREING KEY contacto REFERENCES personal(codP)  
ON DELETE RESTRICT ON UPDATE CASCADE  
);
```

## Implementación SQL (1)

```
CREATE TABLE actividad(  
  codActividad SMALLINT,  
  nomActividad VARCHAR(30) NOT NULL,  
  descripcion VARCHAR(50) NOT NULL,  
  nivel SMALLINT,  
  PRIMARY KEY (codActividad)  
);
```

```
CREATE TABLE habitacion(  
  numeroHab SMALLINT,  
  nomHotel SMALLINT,  
  tipo TINYINT NOT NULL,  
  banno BOOLEAN,  
  precio NUMERIC(6,2),  
  PRIMARY KEY (nomHotel, numeroHab),  
  FOREIGN KEY (nomHotel) REFERENCES hotel(num)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE  
);
```

```
CREATE TABLE realizaActividad(  
  idActividad SMALLINT,  
  idHotel SMALLINT,  
  diaSemana VARCHAR(10),  
  PRIMARY KEY (idActividad, idHotel),  
  FOREIGN KEY (idHotel) REFERENCES Hotel(num)  
  ON DELETE CASCADE ON UPDATE CASCADE,  
  FOREIGN KEY (idActividad) REFERENCES ACTIVIDAD(codigo)  
  ON DELETE CASCADE ON UPDATE CASCADE  
);
```

**Si se elimina o se modifica un hotel, se deben eliminar o modificar las habitaciones de ese hotel**

## Modelo generado con la opción ingeniería inversa de Workbench

Es generado una vez que se ha realizado todo el proceso incluidas las instrucciones SQL de creación de la base de datos

