

# Teo: Diseño de Esquema de BD de Calidad

☒ Archive



[Introducción](#)

[1. Redundancia de Datos](#)

[2. Relaciones Implícitas](#)

[3. Uso Incorrecto de Atributos en Relaciones](#)

[4. CR de Grado  \$> 2\$  \(vs\) CR binarias](#)

[5. CE Débiles vs Atributo Multivalorado Compuesto](#)

[CEs vs Atributos](#)

[Uso de CE vs CR](#)

## Introducción

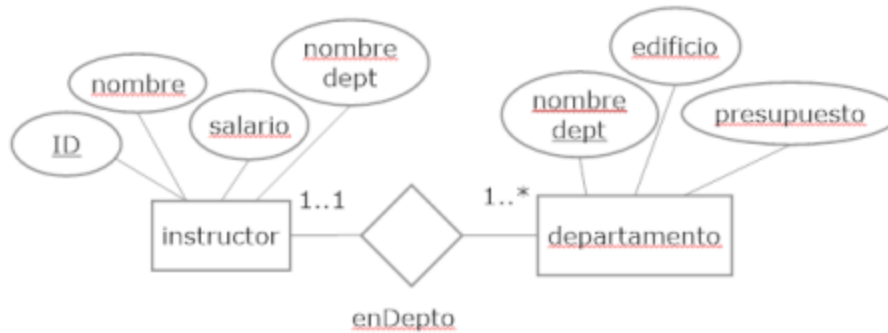
Veremos algunas pautas para modelar esquemas de BD (o sea, modelado de diagramas ER) de calidad basándonos en ciertos **criterios** importantes:

- Redundancia de datos
- Comprensibilidad
- Completitud
- Facilidad de consulta.

## 1. Redundancia de Datos

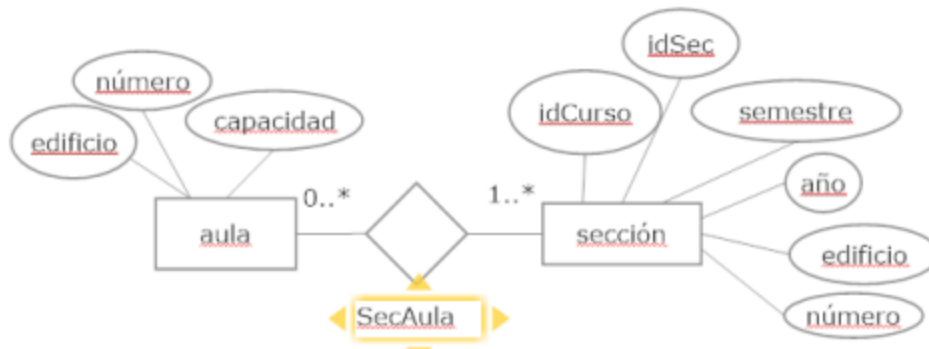
Es importante que nuestro modelo ER no contenga redundancias en los atributos.

Ejemplo 1:



*nombre dept* es la clave primaria de *departamento* y es un atributo que también aparece de forma redundante en *instructor*, debemos removerlo.

Ejemplo 2:



*edificio* y *número* son redundantes en *sección*.

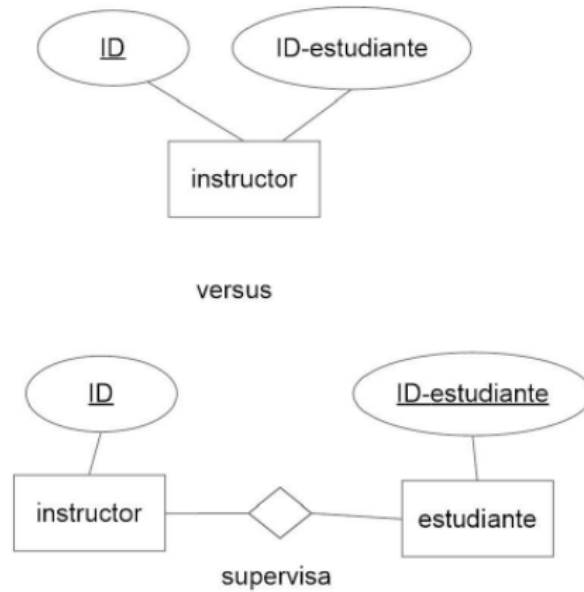


Los atributos redundantes implican costos de almacenamiento extra y propagación de modificaciones innecesarias.

## 2. Relaciones Implícitas

A la hora de diagramar, debemos explicitar las conexiones que existan entre entidades distintas a través de **relaciones**. No debemos usar atributos que de forma implícita representen este tipo de situaciones.

Ejemplo:

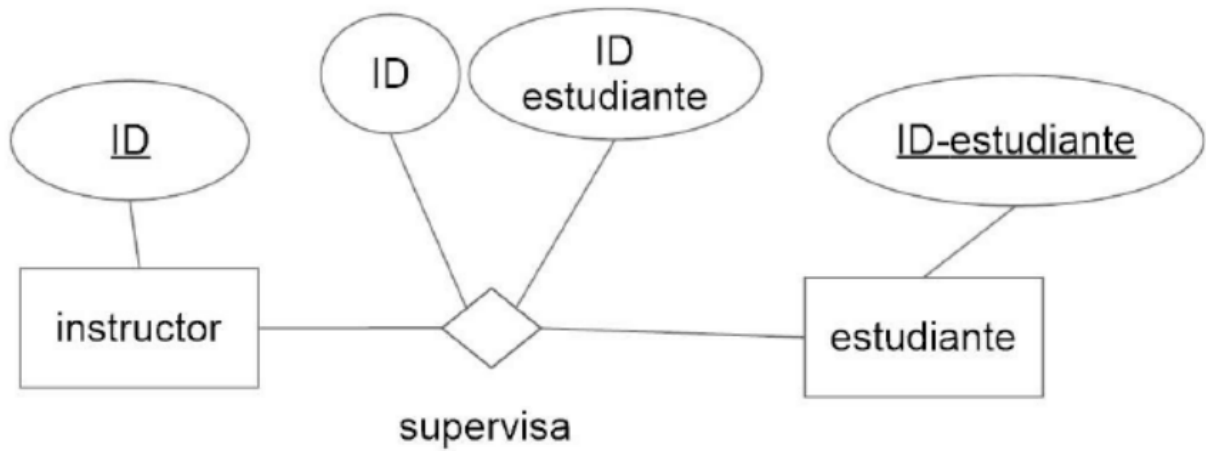


En este caso, conviene el segundo diagrama porque ID-estudiante esta diciendo de forma implícita que hay una relación entre instructor y estudiante, por lo que conviene utilizar **relaciones** para representar esto.

### 3. Uso Incorrecto de Atributos en Relaciones

No es correcto usar las claves primarias de entidades que se relacionan entre sí, como atributos de su relación.

Ejemplo:



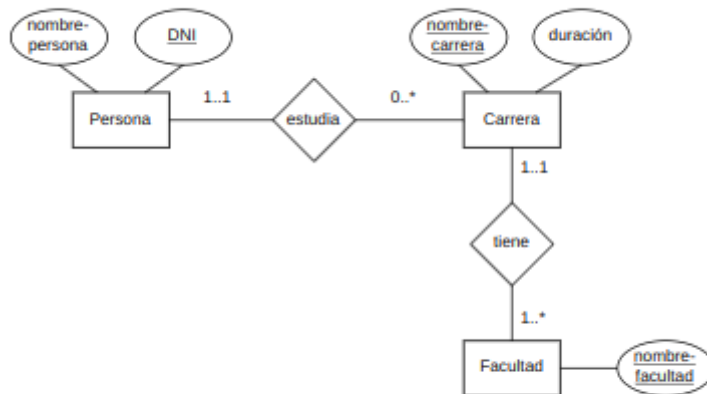
## 4. CR de Grado > 2 (vs) CR binarias

A la hora de decidir que grado utilizar en los conjuntos de relaciones, debemos tener en cuenta:

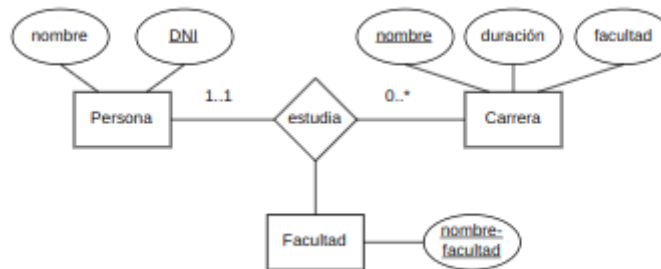
- Mantener diseños simples y fácil de entender
- Fáciles de gestionar
- Fácil de aplicar restricciones de integridad (claves primarias etc)

Ejemplo:

### Grado 2



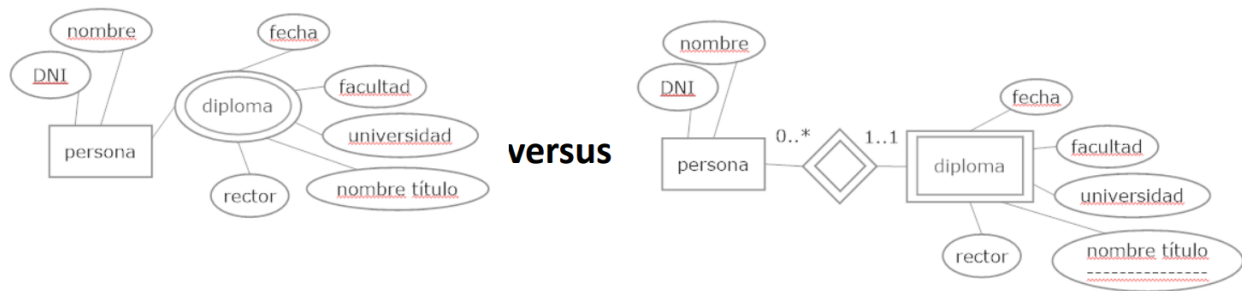
### Grado 3



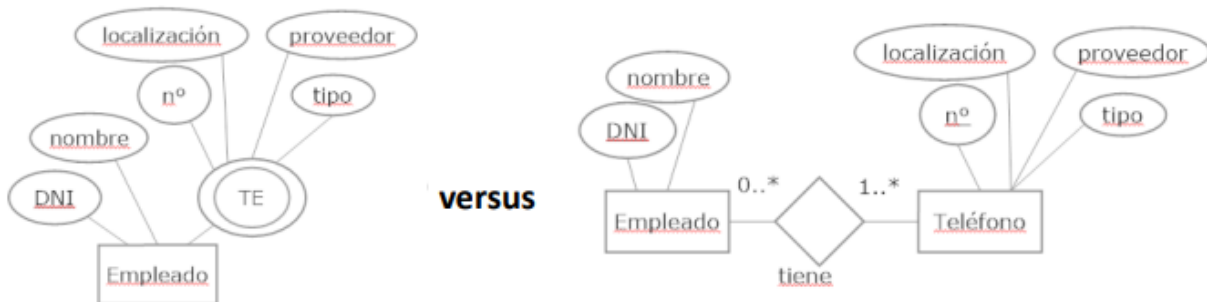
En este caso, el mas conveniente es el binario (binario porque toda relación une siempre a dos conjuntos de entidad) ya que captura los datos del CR de grado > 2 y puede expresar más restricciones de integridad (más claves primarias, etc).

## 5. CE Débiles vs Atributo Multivalorado Compuesto

Puede pasar que existan entidades con muchos atributos, volviendo a la entidad demasiado compleja. En estos casos, conviene hacer un conjunto de entidad débil.



## CEs vs Atributos



Si no sabemos modelar algo como una entidad o un atributo, pensemos:

- Esta cosa puede tomar una sola forma y existe una sola? Entonces como atributo simple.
- Esta cosa puede tomar una sola forma y existen varias? Entonces como atributo multivalorado.
- Esta cosa puede tomar varias formas y existe una cantidad determinada + puede ser poseída por una entidad? Entonces como entidad (porque puedo usar cardinalidades).
- Esta cosa puede tomar varias formas y existe una cantidad determinada + puede relacionarse con varias otras entidades? Entonces como entidad (para eliminar redundancias).

## Uso de CE vs CR

Para decidir cual de las opciones es la mejor en el diseño de bases de datos, es importante tener en cuenta la cardinalidad de la relación y si existen atributos adicionales relacionados con la conexión entre entidades.

Ejemplos:

1. Naturaleza de los datos:

- Si el objeto que se quiere representar tiene atributos propios y distintos no complejos, entonces conviene modelarlo como una entidad.
- Si queremos representar una relación entre dos o más objetos, y no hay atributos específicos para la relación entre sí, conviene modelarlo como relación.

2. Cardinalidad de la relación

- Si la relación es 1..1 o 1..N, a menudo es conveniente representarla como una entidad. Si es varios a varios, es necesario utilizar una relación. (filmina difiere acá)

3. Atributos adicionales

- Si la relación entre objetos tiene atributos adicionales como una fecha de inicio o un estado, entonces conviene usar una relación para almacenar esos detalles.

Conclusiones de las filminas:

- Cuando el CR es uno-varios o varios-uno y hay atributos no de clave primaria en el CE del lado uno, **conviene la opción CR**.
- Si el CR es varios-varios y en al menos en un CE hay atributos que no son de clave primaria, **conviene la opción CR**.