

Master Universitario en Inteligencia Artificial Aplicada

Bases de Datos e Infraestructuras

uc3m

Universidad
Carlos III
de Madrid



0.- Introducción

1.- Diseño de BB.DD. Relacionales (estática)

2.- Operación de BB.DD. Relacionales (dinámica básica)

3.- SQL para consultas analíticas

4.- Otros elementos de las BBDD Relacionales

Estructura del Tema

- 1.- Introducción y Conceptos Básicos**
- 2.- Definición de Relación y Ejemplos**
- 3.- Implementación de una Relación (SQL)**
- 4.- Restricciones en una relación**
- 5.- Asociación de Relaciones**
- 6.- El Diseño Relacional**

- Es un **Modelo de Datos** (herramienta para describir archivos de datos)
- Se basa en la noción matemática de relación:
$$R: N \times N, \quad R \{(x, y) / y = x^2\}$$
$$R \{(1, 1), (2, 4), (3, 9), \dots\}$$
- De modo análogo, los datos relevantes de una relación describen las ocurrencias que ésta tiene en cualquier momento del tiempo



persona (1234, 'Fulano', 28, 551234, 'fulano@email...')
persona (5678, 'Mengano', 82, 5554321, 'mengano@email...')
persona (...)
coche ('1234ABC', 'Seta', 'Basic', '1-1-2011', 0)
coche ('1235ABC', 'Seta', 'Basic', '1-1-2011', 100000)
coche (...)

OCURRENCIAS

Dominio: conjunto de valores de la misma naturaleza

Ejemplo: *numeros_teléfono* son números naturales de nueve dígitos ($\geq 9 \cdot 10^8$, $< 10^9$)

Relación: subconjunto del producto cartesiano de n dominios

persona: $DNI \times Nombres \times numeros_telefono$

persona: {(03456789, José, 555112233), (03451940, Juan, 555123123), ...}

Atributo: propiedad común a los elementos de una relación.

Se define sobre un dominio (y su valuación se restringe al mismo).

Recibe un solo valor para cada fila.

*Dominios, relaciones, y atributos concretos deben ser **nominados** de modo significativo (semánticamente) con etiquetas identificativas (no pueden repetirse, en el mismo ámbito; sí en distintos ámbitos)*

Esquema de una Relación:

asociación de atributos que **caracteriza** y distingue a los miembros de una relación
COCHE (Matrícula, N°Serie, Marca, Modelo, Año)

Ocurrencia de una Relación (*fila, registro o tupla*):

asociación de valores (uno por atributo) referidos a un esquema de relación, que mantiene una correspondencia biunívoca con un individuo en el mundo real (perteneciente a la generalidad que representa la relación).

Ejemplo: $\langle 739ABD, A69352, Ford, Fiesta, 1992 \rangle$ se refiere a un *coche* real

Representación y notación:

- de un Esquema de Relación: nombre de la relación seguido de la colección de atributos que la caracterizan (el orden no importa)
- de una Relación en tabla: una columna por atributo, una fila por tupla (el orden de las columnas no importa, pero es el mismo en todas las filas).

“ Una empresa reparte paquetes mediante drones.

El paquete tiene un código de barras, dimensiones, y peso.

El cliente tiene nombre, DNI, teléfono, dirección, provincia y país.

Los drones tienen número de serie, potencia, y carga máxima, y son los que llevan los paquetes.”

RELACIÓN (atributo1, atributo2, atributo3, ...)

...

...

“ Una biblioteca tiene libros, revistas y soportes de información digital.

Los libros tienen título, autor, año publicación, editorial, y número de páginas.

Las revistas tienen nombre, número, fecha, precio, editorial, y número de páginas.

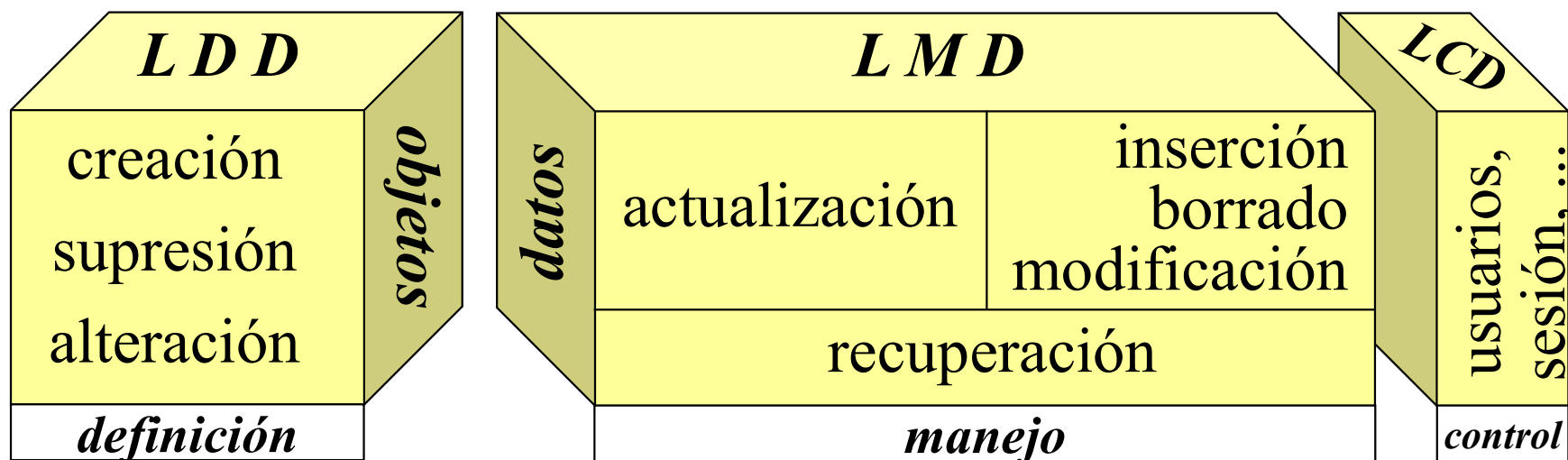
Los soportes tienen título, tipo soporte, tipo contenido, duración, y fecha”.

RELACIÓN (atributo1, atributo2, atributo3, ...)

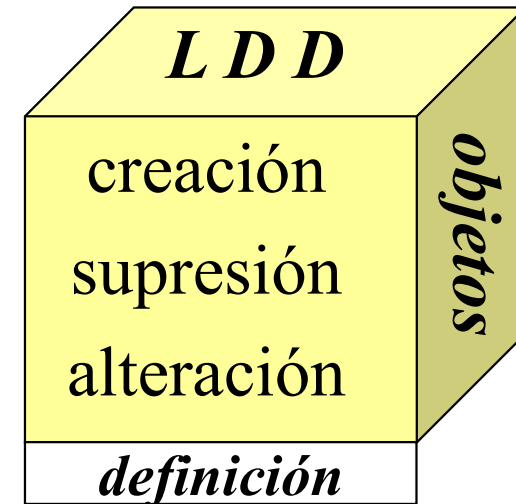
...

...

- SQL contiene los tres tipos de lenguajes de datos
 - Lenguaje de Definición de Datos
 - Lenguaje de Manipulación de Datos
 - Lenguaje de Control de Datos



- Tres acciones básicas:
 - Creación (CREATE)
 - Supresión (DROP)
 - Alteración (ALTER)
- Sobre:
 - Esquemas
 - Tablas
 - Vistas
 - Dominios
 - Secuencias...



```
CREATE TABLE nombre_tabla (  
    elemento_de_tabla,  
    elemento_de_tabla,  
    [...]  
);
```

elemento_de_tabla := { columna | restricción_tab }

columna := *nombre_columna* **datatype**(**max_size**)
[**val_column**] [**restricción_columna**]

- Se pueden clasificar por su:
 - Implementación: Nativos, de Usuario y Externos (paquete, etc)
 - Naturaleza: Numéricos, Alfanuméricos, Fechas y Binarios
 - Operabilidad: comparables, agregables, ...

DATATYPES	Descripción	En Oracle
Numéricos	Número natural Número real de coma fija Número real de coma flotante (5 bytes) Número real de coma flotante (9 bytes)	NUMBER(n) NUMBER(n,p) BINARY_FLOAT BINARY_DOUBLE
Alfanuméricos	Cadena de caracteres longitud fija Cadena de caracteres longitud variable Cad. Caracteres Unicode long. variable	CHAR(n) VARCHAR2(n) NVARCHAR2(n)
Tiempo	Fecha y Hora (7 bytes) Fecha, hora, y fracción de seg. (11 bytes)	DATE TIMESTAMP
Binarios	Binario 'crudo', texto indefinido, ... objetos grandes ...	RAW / LONG / LONG RAW ... BLOB / CLOB / NCLOB / BFILE

2. Supresión:

```
DROP <elemento> <nombre>;
```

```
TRUNCATE <elemento> <nombre>;
```

3. Alteración:

```
ALTER <elemento> <nombre>  
    {ADD|ALTER|DROP} <elemento>  
    [<definición>;
```

EJEMPLOS:

```
ALTER TABLE ejemplo ADD (edad NUMBER(2));
```

```
ALTER TABLE ejemplo DROP COLUMN max_edad;
```

```
ALTER TABLE ejemplo SET UNUSED max_edad;
```

```
ALTER TABLE ejemplo DROP UNUSED COLUMNS;
```

```
DROP TABLE persona;
```

Son limitaciones que se imponen a la valuación de los atributos

Distinguiremos los siguientes tipos:

- Opcionalidad / Obligatoriedad
- Univocidad
- Restricciones (simples) de Rechazo
- Referencias (asociación entre relaciones)

Opcionalidad: el atributo opcional puede no adoptar ningún valor del dominio sobre el que se define por distintas causas:

- El valor se desconoce (no ha sido facilitado)
- El valor no existe aún (ejemplo, *fecha_defunción*)
- El atributo no es aplicable a esa tupla (el atributo *carburador* de la relación *coche* no es aplicable a coches tipo ‘diesel’)

Notación: los atributos opcionales se marcan con un asterisco*

Valor Nulo: es el valor que adopta el atributo opcional en esos casos

- observar la diferencia entre *null* y el valor 0 ó el valor ‘’

Clave: conjunto de atributos con una función definida

Ejemplos: clave de búsqueda, clave de ordenación, ...

Notación: si tiene varios atributos, pueden agruparse con llave horizontal

superclave: clave capaz de **identificar** unívocamente cada tupla.

clave candidata: superclave mínima

- *minimalidad*: no existe ningún subconjunto suyo capaz de desempeñar la misma función (en este caso, ser clave candidata)

clave primaria: clave candidata privilegiada; se representa **subrayada**.

clave secundaria (alternativa): lo son el resto de claves candidatas.

Si se necesita alguna, debe subrayarse de modo discontinuo.

Se trata de expresiones condicionales sobre los datos que afectan sólo a la fila que está siendo operada

- no debe afectar ni al resto de filas ni al resto de tablas

Ejemplos: *tacha los que te parezcan erróneos*

- el atributo edad tiene un valor menor que o igual a 120
- el atributo ciudad está registrado como válido en la tabla *ciudades*
- el *turno* puede ser “mañana”, “tarde” o “noche”
- el atributo edad tiene un valor comprendido entre 18 y 67
- el atributo vivo admite los valores “SI” y “NO”
- no existen más de 40 alumnos por curso (mismo valor en *grupo*)
- si existe *fecha_defunción*, esta es posterior a *fecha_nacimiento*
- longitud máxima de *Nombre* es 25, la de *Apellido* 20, y entre ambas 40


```
CREATE TABLE nombre_tabla (  
    elemento_de_tabla,  
    elemento_de_tabla,  
    [...]  
);
```

elemento_de_tabla := { *columna* | *restricción_tab* }

columna := *nombre_columna* **datatype**(**max_size**)
 [*valuación_auto*]
 [*restricción_columna*]

valuación_auto := { **DEFAULT** *valor*
 | **GENERATED** {**ALWAYS**|**BY DEFAULT**} **AS IDENTITY**
 [(**START WITH** *int* [, **INCREMENT BY** *int*])] }

RESTRICCIONES:

- de columna

[CONSTRAINT nombre]

[NOT] NULL
PRIMARY KEY
UNIQUE
REFERENCES...
CHECK (...)

[estado]

ENABLE
 DISABLE

- de tabla

[CONSTRAINT nombre]

PRIMARY KEY (col[,...])
UNIQUE (col[,...])
FOREIGN KEY (col[,...]) REFERENCES ...
CHECK (expresión_condicional)

[estado]

No existen vinculaciones físicas. Sólo lógicas.

- Para interrelacionar tuplas, habrá que apuntar desde una tupla a otra. Ese apuntamiento se hará en el ámbito lógico, es decir, incluyendo en una relación un atributo capaz de identificar tuplas en la otra.
- Por tanto, los atributos referenciados serán una **superclave** (habitualmente se utiliza la clave primaria, por ser privilegiada)
- Por otro lado, los atributos referenciantes se denominan *clave ajena*

Ejemplo: el atributo *dueño* en la relación *coches* adopta *DNI* de *personas*

- **Notación**: se representan con un arco dirigido (flecha) desde la clave ajena hasta la superclave referenciada. Si se referencia a la clave primaria, se puede representar apuntando al nombre de la relación.

Cardinalidad de la asociación: número de tuplas que intervienen (cantidad máxima, por parte de cada relación).

- 1 a 1: correspondencia biunívoca

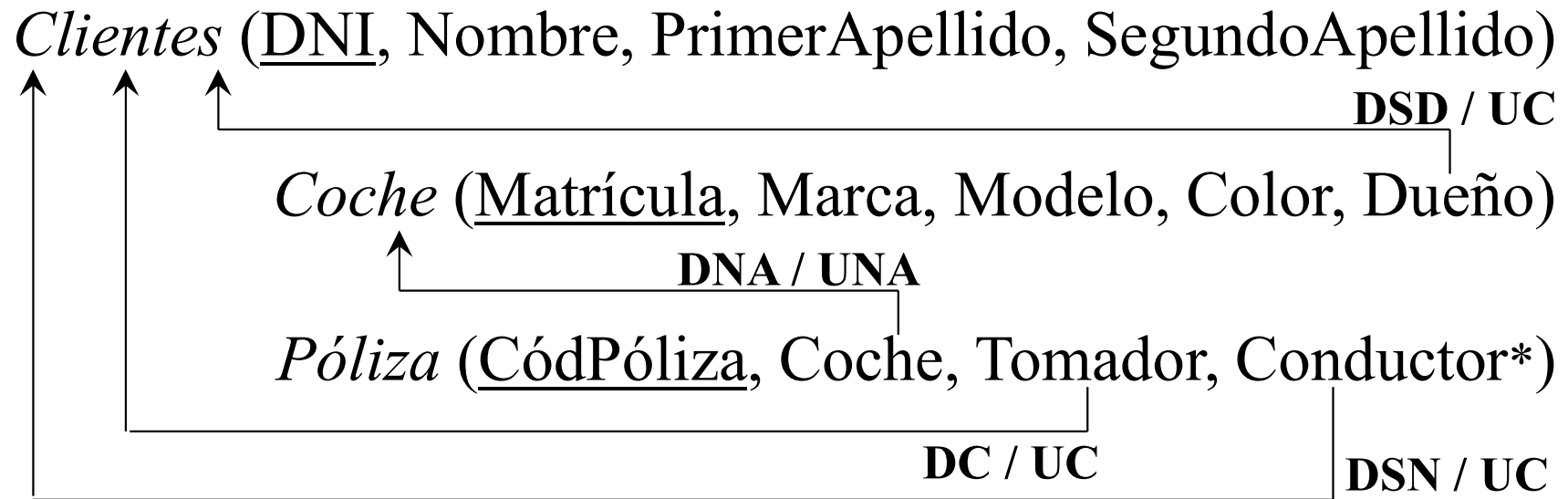
Se podría fusionar ambos esquemas en uno sólo. La clave ajena puede localizarse en cualquiera de los dos (o bien compartir clave primaria)

- 1 a n: correspondencia múltiple

Un esquema (1) es padre del otro (n).

La clave ajena se sitúa en la relación que participa con múltiples tuplas (propagación de clave).

- n a n: tiene que existir **una relación intermedia**.



¿Puedes leer este diseño?

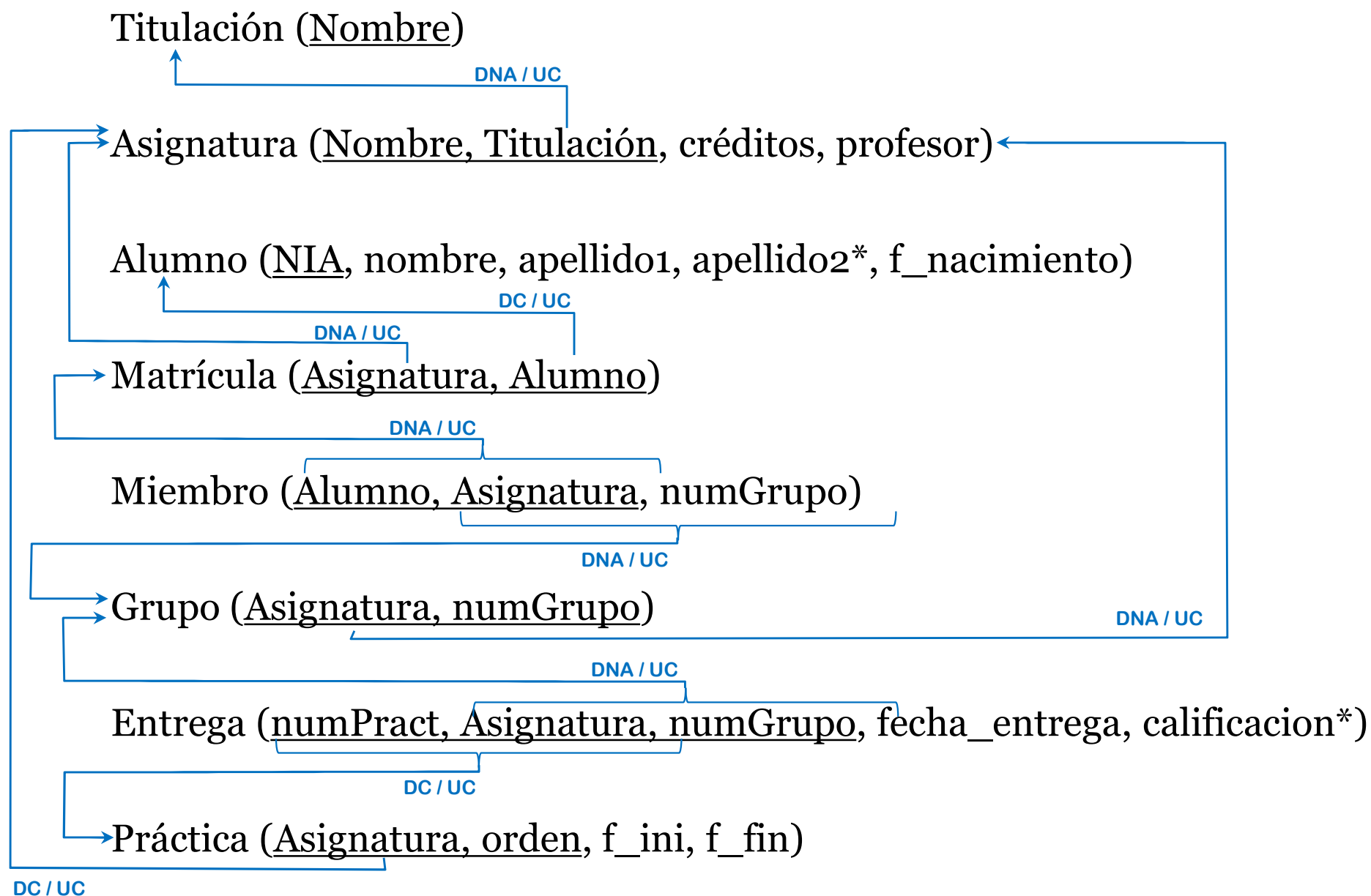
Señala V/F en estas afirmaciones

- Un coche puede tener varias pólizas de seguros.
- Todo coche tiene al menos una póliza de seguros.
- Varios coches pueden compartir una póliza de seguros.
- El tomador y el conductor en una póliza son el mismo cliente.
- Una póliza no existe sin tomador, pero puede carecer de conductor.

Acciones de *rechazo* o de *corrección automática*, aplicadas para evitar pérdidas de integridad referencial (valores no válidos en claves ajenas)

- **Restrict** (operación restringida): **R**
si la operación involucra padres con hijos, no se lleva a cabo
- **No Action** (operación cancelada): **NA**
si la operación rompe la integridad referencial, no se lleva a cabo
- **Cascade** (propagación en cascada): **C**
los valores no válidos serán actualizados también en la clave ajena
- **Set Null** (puesta a nulo): **SN**
los valores no válidos serán sustituidos por el valor nulo
- **Set Default** (valor por defecto): **SD**
los valores no válidos serán sustituidos por un valor por defecto

- Se necesita una base de datos para gestionar las prácticas:
 1. Debe recogerse la información de los alumnos (NIA, nombre, primer apellido y opcionalmente segundo apellido, edad y fecha de nacimiento)
 2. También hay asignaturas: nombre, créditos, titulación, profesor
 3. Debe recogerse la formación de grupos de prácticas. Los grupos tienen un identificador numérico (3) distinto para cada asignatura, y los forman alumnos matriculados en la asignatura.
 4. Las prácticas de cada asignatura se identifican por su núm_orden y tienen enunciado, plazo (fecha inicio, fecha fin), y calificación para cada grupo.
 5. Un alumno no puede cancelar matrícula mientras sea miembro de un grupo. Si un alumno es expulsado (eliminado de la base) se cancelan todas sus matrículas. Si una práctica desaparece, también lo hacen sus entregas. En el resto de los vínculos, no deben permitirse borrados que dejen registros huérfanos. Todas las modificaciones se propagan.




```
CREATE TABLE nombre_tabla (  
    elemento_de_tabla,  
    [...]  
);
```

elemento_de_tabla := { columna | restricción_tab }

columna := *nombre_columna* datatype(max_size)
[val_column] [restricción_columna]

restricción_tab := [CONSTRAINT nombre]
{
 PRIMARY KEY (col[,...])
 |
 UNIQUE (col[,...])
 |
 FOREIGN KEY (col[,...]) REFERENCES ...
 |
 CHECK (...)
}

Clave ajena: es una restricción (a nivel de tabla o de columna)

FOREIGN KEY (<columnas>)

REFERENCES <nombre_tabla>

[(<columnas>)]

[ON DELETE <acción>]

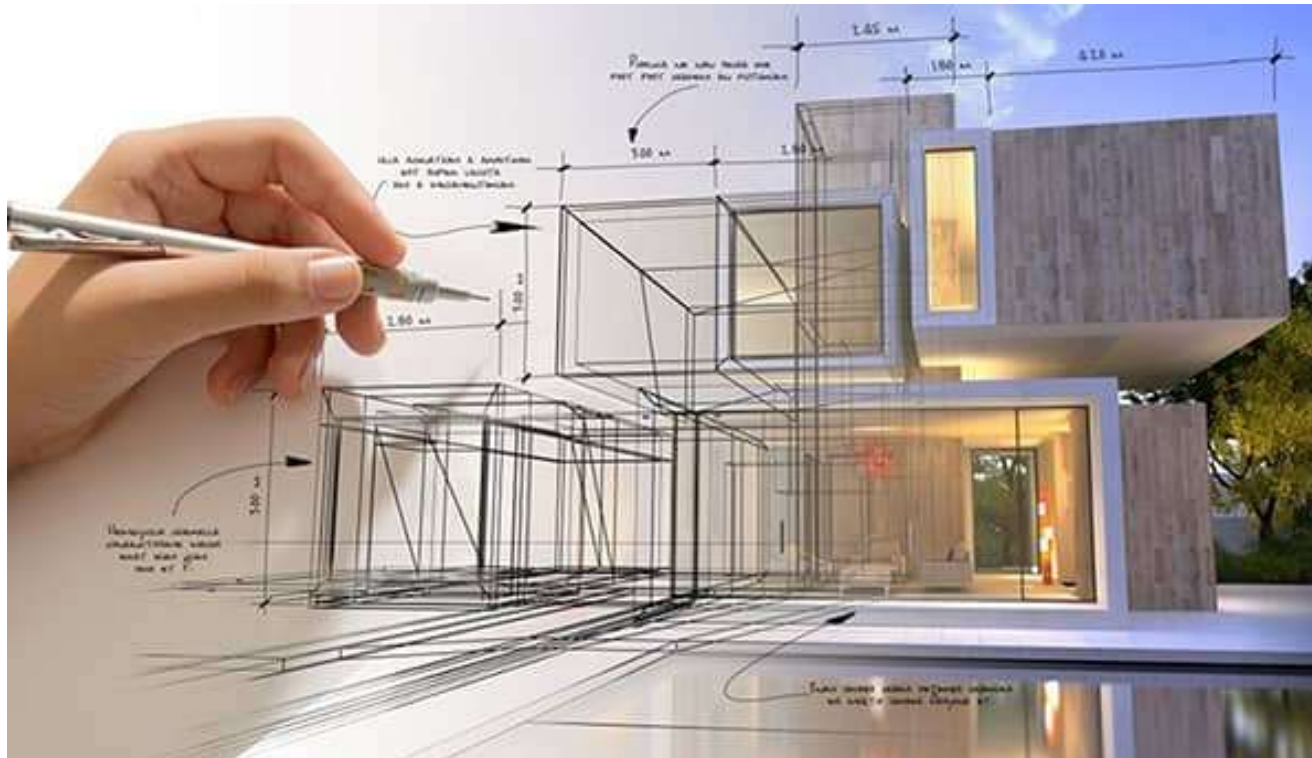
[ON UPDATE <acción>]

- Oracle permite las siguientes opciones de IR:
 - Actualización: por omisión, aplica **NO ACTION**
 - Borrado: omisión **NO ACTION** | **CASCADE** | **SET NULL**

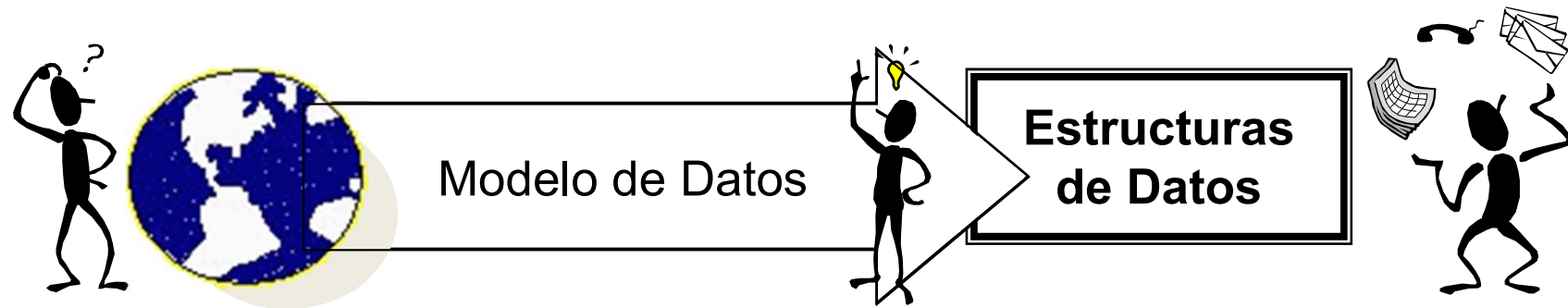
```
CREATE TABLE ejemplo(  
    identifier NUMBER(3) ,  
    padre NUMBER(3) ,  
    titulo VARCHAR2(50) NOT NULL,  
    cantidad NUMBER(8)  DEFAULT(8) ,  
    CONSTRAINT pk_ejemplo PRIMARY KEY(identifier) ,  
    CONSTRAINT fk_ejemplo FOREIGN KEY (padre)  
        REFERENCES ejemplo(identifier) ON DELETE CASCADE ,  
    CONSTRAINT unigenito UNIQUE(padre) ,  
    CONSTRAINT ttl_valido CHECK titulo IN ('A','B','C')  
);
```

```
DESC ejemplo
```

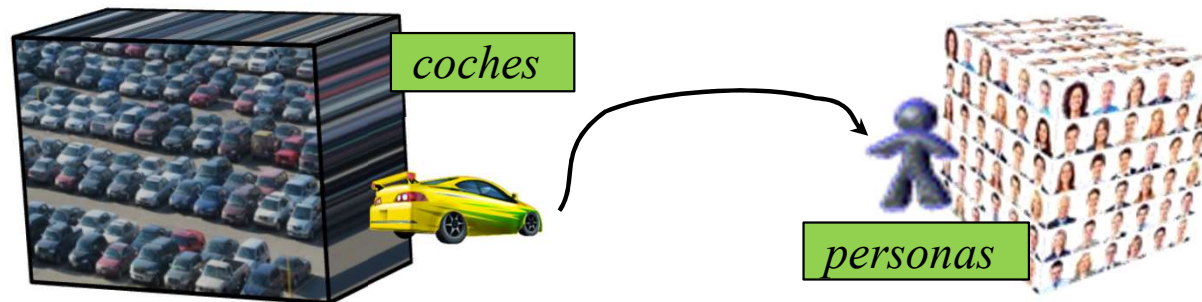
*Proceso de análisis de las necesidades de información,
concepción, definición y descripción de la solución
mediante una Base de Datos*



Se requieren herramientas conceptuales y descriptivas



Estática del Modelo de Datos: *objetos, asociaciones y restricciones*



- “Las personas no se repiten”
- “Un solo coche por persona”
- La edad de persona < 150
- Si un coche tiene dueño, esa persona existe

Restricciones: Inherentes vs. Semánticas

- Es un **Modelo de Datos** (herramienta para describir archivos de datos)
- Tiene tres elementos básicos: dominio, **relación** y atributo.
- Una relación se puede describir con un esquema de relación:
 - nombre de la relación seguido de sus atributos
 - encerrados entre paréntesis y separados con comas
 - subrayando su clave primaria
 - y sus claves alternativas (subrayado discontinuo),
 - y marcando con un asterisco los atributos opcionales



PERSONAS(DNI, nombre, edad, teléfono, email)

COCHES(Matrícula, marca, modelo, matriculación, km)

Los Datos



necesitan semántica



propósito



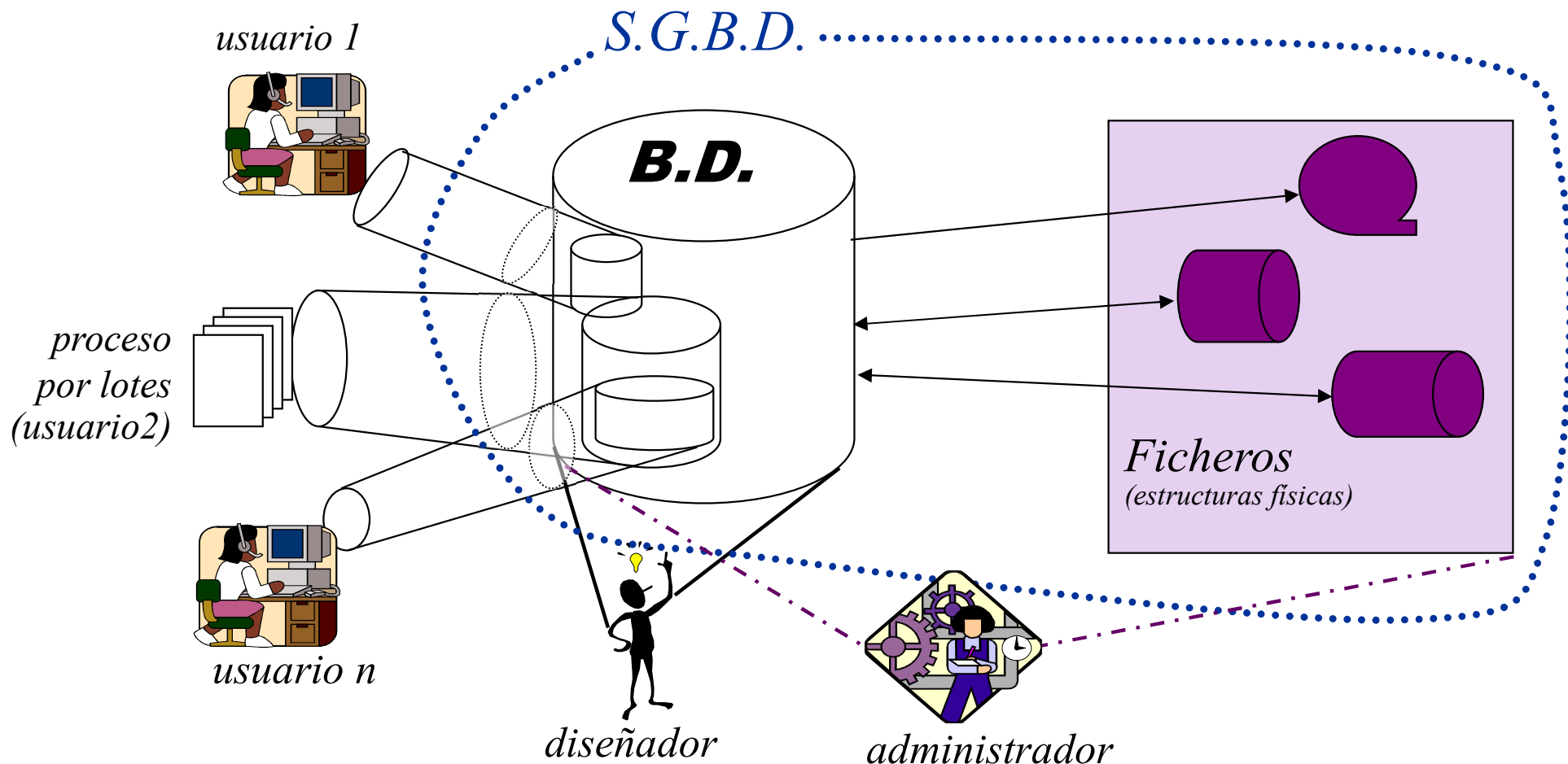
y oportunidad



*¿ Cómo se gestionan los datos ? ¿ Dónde se almacenan ?
¿ Quién los “mueve” ?*



*Conjunto coordinado de herramientas que proporciona los medios necesarios para **interaccionar** con la base a **todos** los niveles*



Funciones esenciales de un SGBD

- Descripción

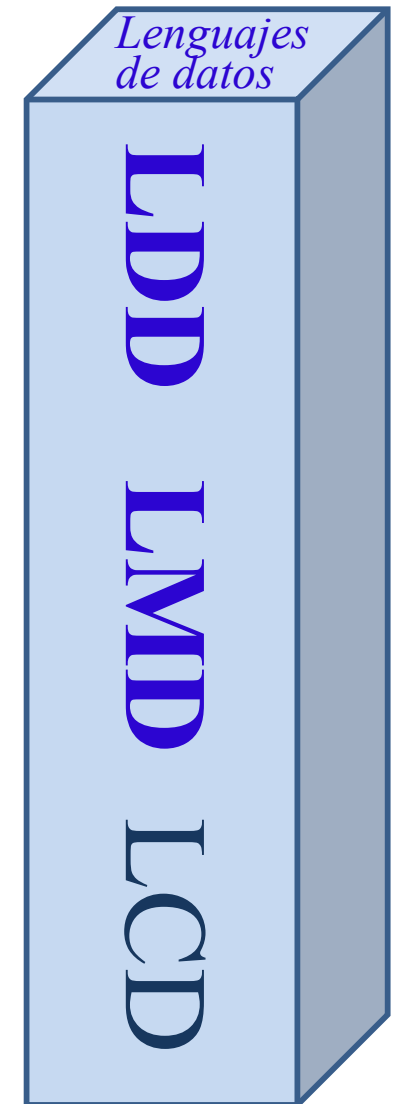
Permite definir los elementos de datos y su estructura, las interrelaciones entre ellos y las reglas de validación semántica.

- Manipulación

Posibilita la operación del contenido de la base

- Utilización

Herramientas para la administración de la BD.



- El conjunto de todos los esquemas de relación de una BDR, con sus interrelaciones y RI que aplican, se denomina **Esquema Relacional**. El diagrama que utilizamos para describirlo se llama Grafo Relacional.
- El código generado para crear la BD y los paquetes que recogen todos los procedimientos para su operación, se registra en ficheros de texto (plano) denominados scripts (sql).
- Las fases más usuales en el desarrollo de una BD son:



En cada una de ellas, el producto es “Documentación de Proyecto”.
En algunas, también se obtienen scripts (código sql) como resultado.

“Depósito de datos integrados, consistentes, definidos y estructurados”

- *Colección o almacén de datos integrados*
- *... con redundancia controlada*
- *... cuya estructura refleja las interrelaciones y restricciones del mundo real*
- *... cuyos datos serán independientes de aplicación o usuario*
- *y tendrán definición y descripción únicas (y almacenada con ellos).*
- *... y cuyos procedimientos involucrados preservarán la integridad de la Base*
- *respetando además ciertas normas de disponibilidad y confidencialidad*