



BASES DE DATOS (IG18 Semipresencial)

Diseño Físico de Bases de Datos Relacionales.

Lledó Museros / Ismael Sanz
museros@icc.uji.es / isanz@icc.uji.es



1. Introducción
2. Metodología de diseño físico



➤ Fases en el Diseño de BBDD

Especificación de requisitos

DISEÑO CONCEPTUAL

Esquema conceptual

DISEÑO LÓGICO

← Normalización

Esquema lógico

DISEÑO FÍSICO

Esquema físico



1. Introducción
2. Metodología de diseño físico



- **Traducir el esquema lógico global para el SGBD específico.**
 1. Diseñar las relaciones base.
 2. Diseñar las reglas de negocio.
- **Diseñar la representación física.**
 3. Analizar las transacciones.
 4. Escoger las organizaciones de ficheros.
 5. Escoger los índices secundarios.
 6. Considerar la introducción de redundancias controladas.
 7. Estimar la necesidad de espacio en disco.
- **Diseñar los mecanismos de seguridad.**
 8. Diseñar las vistas de los usuarios.
 9. Diseñar las reglas de acceso.
- **Monitorizar y afinar el sistema.**



- **Traducir el esquema lógico global para el SGBD específico.**
 - Diseñar las relaciones base. Para cada relación base (tabla) hay que especificar:
 - nombre,
 - lista de atributos, dominios y valores por defecto,
 - clave primaria y claves ajenas,
 - reglas de integridad para las claves ajenas.

```
CREATE DOMAIN PNUM      AS VARCHAR(5);
CREATE DOMAIN ENUM      AS VARCHAR(5);
CREATE DOMAIN ONUM      AS VARCHAR(3);
CREATE DOMAIN INUM      AS VARCHAR(5);
CREATE DOMAIN CALLE     AS VARCHAR(25);
CREATE DOMAIN AREA      AS VARCHAR(15);
CREATE DOMAIN POBLACION AS VARCHAR(15);
CREATE DOMAIN TIPO      AS VARCHAR(1)
    CHECK(VALUE IN ('A','C','D','P','V'));
CREATE DOMAIN HAB       AS SMALLINT
    CHECK(VALUE BETWEEN 1 AND 15);
CREATE DOMAIN ALQUILER  AS DECIMAL(6,2)
    CHECK(VALUE BETWEEN 0 AND 9999);
```



```
CREATE TABLE inmueble (  
    inum          INUM          NOT NULL,  
    calle        CALLE         NOT NULL,  
    area         AREA,  
    poblacion    POBLACION    NOT NULL,  
    tipo         TIPO          NOT NULL DEFAULT 'P',  
    hab          HAB           NOT NULL DEFAULT 4,  
    alquiler     ALQUILER     NOT NULL DEFAULT 350,  
    pnum         PNUM          NOT NULL,  
    enum         ENUM,  
    onum         ONUM          NOT NULL,  
    PRIMARY KEY (inum),  
    FOREIGN KEY (pnum) REFERENCES propietario  
        ON DELETE no action ON UPDATE cascade,  
    FOREIGN KEY (enum) REFERENCES plantilla  
        ON DELETE set null ON UPDATE cascade,  
    FOREIGN KEY (onum) REFERENCES oficina  
        ON DELETE no action ON UPDATE cascade  
);
```



- **Traducir el esquema lógico global para el SGBD específico.**
 - Diseñar las reglas de negocio. Se pueden definir restricciones:

```
CONSTRAINT inmuebles_por_empleado
    CHECK (NOT EXISTS (SELECT enum
                        FROM inmueble
                        GROUP BY enum
                        HAVING COUNT(*)>10))
```

- O disparadores (triggers):

```
CREATE TRIGGER inmuebles_por_empleado
ON inmueble
FOR INSERT,UPDATE
AS IF ((SELECT COUNT(*)
        FROM inmueble i
        WHERE i.inum=INSERTED.inum)>10)
BEGIN
    PRINT "Este empleado ya tiene 10 inmuebles
    asignados"
    ROLLBACK TRANSACTION
END
```




- **Diseñar la representación física.**
 - Analizar las transacciones:
 - frecuencia,
 - relaciones y atributos que acceden,
 - atributos en el WHERE de las sentencias SQL,
 - atributos en los JOIN de las consultas,
 - restricciones temporales.
 - Escoger las organizaciones de ficheros.
 - Escoger los índices secundarios. Algunas recomendaciones son:
 - crear un índice sobre cada clave primaria,
 - no crear índices sobre relaciones pequeñas,
 - crear índices sobre claves ajenas que se utilizan con frecuencia en JOINS,
 - evitar índices sobre atributos que cambian con frecuencia,
 - evitar índices poco selectivos,
 -



- **Diseñar la representación física.**
 - Considerar la introducción de redundancias controladas. Por ejemplo:
 - combinar relaciones de uno a uno,
 - duplicar atributos no clave en relaciones de uno a muchos,
 - utilizar las tablas de referencia para validar datos,
 - duplicar claves ajenas en relaciones de uno a muchos,
 - duplicar atributos en relaciones de muchos a muchos,
 - introducir grupos repetitivos.
 - Estimar la necesidad de espacio en disco:
 - número de tuplas de cada relación y
 - tamaño de las tuplas.
- **Diseñar los mecanismos de seguridad.**
 - Diseñar las vistas de los usuarios.
 - Diseñar las reglas de acceso.
- **Monitorizar y afinar el sistema....**



BASES DE DATOS (IG18 Semipresencial)

Diseño Físico de Bases de Datos Relacionales.

¿Dudas?

Lledó Museros / Ismael Sanz
museros@icc.uji.es / isanz@icc.uji.es