

WUOLAH



senpereido

www.wuolah.com/student/senpereido



49

Resumen primer parcial.pdf

Resumen primer parcial FBD



2º Fundamentos de Bases de Datos



Grado en Ingeniería Informática



**Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada**

Linguaskill 
from Cambridge

Ya puedes sacarte tu B1/B2/C1 de inglés desde casa

Demuestra tu nivel en 48 horas

Nuevo

#LinguaskillEnCasa

Tema 1: INTRODUCCIÓN Y DEFINICIONES

Base de Datos: Conjunto de datos comunes a un proyecto almacenados sin redundancia para ser útiles a diferentes aplicaciones.

Sistema Gestor de Bases de Datos: Conjunto de elementos software con capacidad para definir, mantener y utilizar una Base de Datos.

Debe permitir:

- Definir estructuras de almacenamiento
- Acceder a los datos de forma eficiente y segura
- Organizar la actualización de los datos y el acceso multiusuario

Operaciones que se realizan en un **SGBD**:

- **Insertar** datos
- **Modificar** datos existentes
- **Borrar** datos existentes
- **Obtener datos** de la BD

Elementos involucrados en una **BD**:

Datos:

- Integrados (sin redundancia)
- Compartidos (varias aplicaciones)

Hardware:

- BD normal o distribuida

Software DBMS (DataBase Management System):

- Programas para describir estructuras y gestionar la información en una BD.

Usuarios:

- Usuario final
- Programador de aplicaciones
- Administrador de la BD

Concepto: **Dato operativo**:

Pieza de información básica que necesita una empresa/proyecto/aplicación para su funcionamiento. Puede ser:

- Entidad
- Atributo
- Relación

Cuando se determinan y clasifican todos los datos operativos, se obtiene el **Esquema Lógico** de la BD.

Concepto: **Independencia**:

Los datos se organizan independientemente de las aplicaciones que van a usarlos.

Independencia física: El diseño lógico de la BD debe ser independiente a todos los niveles del almacenamiento físico de los datos. Esto permite realizar cambios físicos sin alterar la lógica de la aplicación y liberar a las aplicaciones de gestionar aspectos relativos al almacenamiento.

Independencia lógica: Persigue que los cambios en el esquema lógico no provoque que las aplicaciones necesiten ser modificadas.

Con esto se consigue:

- Aumento de seguridad y fiabilidad
- Menos problemas para las aplicaciones

No nos vemos pero ahora estamos más cerca #QuédateEnCasa

Primera academia especializada en estudios de la Facultad de Informática UCM

 academia@mathsinformática.com

 C/Andrés Mellado, 88 duplicado

 www.mathsinformatica.com

 academia.maths

 91 399 45 49

 615 29 80 22

Material online

- ♦ Resolución de ejercicios en vídeo
- ♦ Clases virtuales Skype y Hangouts
- ♦ Grupos de asignatura en Whastssap

Maths 
 **informática**


PROFESORES
TITULADOS


MÁS DE 30 AÑOS
DE EXPERIENCIA


Máximo 16 alumnos
por grupo


Plazas limitadas

- Posibilidad de cambios en los esquemas por parte de desarrolladores de aplicaciones y de administradores.

Objetivos de un SGBD:

- Independencia de los datos
- Diseño y utilización orientada al usuario
- Centralización: Los datos deben gestionarse de forma centralizada e independiente de las aplicaciones
- No redundancia
- Consistencia
- Fiabilidad: Protección contra fallos y mecanismos de recuperación
- Seguridad: Gestión de usuarios y privilegios.

Tema 2: ARQUITECTURA DE UN SGBD

Arquitectura con tres niveles

- Nivel interno (DBA)
 - Nivel conceptual (DBA)
 - Nivel externo (Usuarios)
- + Nivel externo:
- Parte de la BD relevante para cada usuario.
 - Datos operativos que le son de interés
 - Representados de la forma que le interesa
 - Datos calculados a partir de los que hay
- + Nivel conceptual:
- Visión global de los datos
 - Estructura lógica de los datos
 - Todas las entidades, atributos y relaciones
 - Restricciones que afectan a los datos
 - Información semántica
 - Información de seguridad
 - Soporte para cada vista externa
 - Ningún detalle de almacenamiento
- + Nivel interno:
- Representación de la BD en el ordenador
 - Cómo se almacenan los datos
 - Busca rendimiento óptimo del sistema
 - Representa:
 - Estructuras de datos
 - Organizaciones en ficheros
 - Comunicación con el SO para almacenamiento
 - Compresión de datos, cifrado, etc...

Correspondencias entre niveles: Fundamental para el establecimiento de la independencia Lógica y Física

+ Transformación conceptual/interna: Como se organizan las entidades lógicas de nivel conceptual almacenados en nivel interno.

- + Transformación externa/conceptual: Describe un esquema externo en términos del esquema conceptual subyacente. (NO SIEMPRE ES POSIBLE)
- + Transformación externa/externa: Algunos SGBD's permiten describir esquemas externos en función de otros esquemas externos.

Lenguajes de una BD: ANSI/SPARC

- + Sublenguaje de datos: DSL
 - Implementado en el propio SGBD
 - Distintas partes
 - Data Definition Language (DDL)
 - Data Control Language (DCL)
 - Data Manipulation Language (DML)
 - Ejemplo destacado: SQL

Enfoques para la arquitectura de un SGBD:

- + Centralizada
 - Problema: Elevado coste
 - Solución: Aproximación Cliente/Servidor
- + Distribuida
 - Problema: Alto coste de mantenimiento
 - Solución: Separar en aplicaciones
- + Arquitectura en tres niveles (Actualmente)
 - Nivel de servidor de datos (distribuido)
 - Nivel de servidor de aplicaciones
 - Nivel de cliente
 - Ventajas: Reducción de costes, mayor facilidad y flexibilidad
 - Inconvenientes: Mayor complejidad

El Administrador de una BD

- + Elaboración del esquema conceptual
 - Análisis de necesidades de la empresa
 - Identificación de datos operativos
 - Elaboración de esquema lógico
 - Implantación de esquema conceptual
- + Decidir estructura de almacenamiento en nivel interno
 - Esquema interno
 - Correspondencia conceptual/interna
- + Conexión con usuarios:
 - Análisis de requerimientos
 - Diseño lógico
 - Codificación con el esquema externo
- + Definir restricciones de integridad
 - Establecer reglas
 - Incluir la integridad en el esquema conceptual
- + Definir e implantar la política de seguridad
 - Gestión de usuarios y privilegios

Formación
Online
Especializada

Clases Online
Prácticas
Becas

Escuela de
LÍDERES

Jose María Girela
Bim Manager.



- + Definir e implantar la estrategia de recuperación frente a fallos
 - SGBD's redundantes
 - RAID
 - Realizar copias de seguridad
 - Política de gestión de transacciones
- + Optimización del rendimiento
 - Liberar espacio
 - Reorganizar operaciones para mayor rapidez
 - Determinar necesidad de nuevo hardware
 - Establecer prioridades en uso de recursos
- + Monitorizar el SGBD
 - Seguimiento de actividad del sistema

Tema 3: Modelos de datos

- Mecanismo formal para representar y manipular información de manera general y sistemática

- Consta de:
 - + Notación para describir datos
 - + Notación para describir operaciones
 - + Notación para describir reglas de integridad

El modelo lógico de datos se obtendrá tras el proceso de análisis de datos y la obtención del esquema conceptual y mediante un proceso de diseño.

- Necesidad de modelos de datos:
 - + Cada esquema se describe con un DDL
 - + Lenguaje de muy bajo nivel
 - + Necesarios mecanismos de más alto nivel para describir los datos sin ambigüedad
- Objetivo:
 - + Describir modelos que representen los datos y los describan de forma entendible y manipulable.

- Clasificación:
 - + Basados en registros (Niveles externo y conceptual)
 - Ejemplos: Modelo jerárquico, modelo en red, modelo relacional
 - + Basados en objetos (Niveles externo y conceptual)
 - + Físicos (Nivel interno)

Modelo Jerárquico

Fue el primero en implementarse físicamente. El nivel externo se compone de aplicaciones Cobol. El gran fallo es que carecía de lenguaje de consultas.

La estructura de datos básica es en árbol, y la BD es una colección de instancias de árboles. Esta estructura permite plasmar directamente las relaciones 1:n y 1:1. Para las relaciones n:m hay que duplicar la información

Inconvenientes:

- Almacenar árboles en ficheros es complejo
- El DML es difícil de usar, así como de implementar

Los registros de tipo secundario tienen dependencia existencial obligatoria con los de tipo raíz

Redundancia para plasmar relaciones n:m, lo que dificulta mantener la integridad de datos

Modelo en Red

La estructura de datos usada es de grafos:

- Nodos: registros
- Arcos: enlaces entre registros
- Relaciones entre conjuntos de entidades

Cualquier registro se puede relacionar con cualquier registro. La BD es una colección de instancias de grafos. La estructura permite plasmar todo tipo de relaciones, especialmente las n:m que pueden ser representadas directamente.

Ventajas

- Estructura más homogénea
- Permite insertar entidades nuevas en un conjunto de forma independiente

Inconvenientes

- La existencia de enlaces entre registros hace que las operaciones DDL y DML sean complejas tanto de implementar como de usar

Modelo Relacional

Representa los datos en forma de tablas o relaciones

BD Relacional: Colección de tablas, cada una de ellas con un nombre único

Conceptos

- Esquema: Esquemas de relaciones junto con restricciones de integridad
- Instancia: Colección de instancias de relaciones que verifican las restricciones de integridad
- BD Relacional: Instancia de una BD junto con su esquema
- Superclave: Conjunto de atributos que identifica unívocamente a cada tupla de una

relación

- Clave candidata: superclave minimal.

De entre las claves candidatas hay que elegir una como principal, llamada **clave primaria**.

Tema 4: El modelo relacional

El modelo relacional abarca tres ámbitos distintos de los datos:

1. Las estructuras para almacenarlos: en forma de tablas
2. La integridad: las tablas preservan la coherencia y la integridad de la información
3. Consulta y manipulación: Los operadores se aplican sobre tablas y devuelven tablas.

La tabla es la estructura lógica del modelo relacional. A nivel físico se almacenan los datos en el formato más adecuado.

Conceptos

Atributo: Elemento susceptible de tomar valores

Dominio: Rango de valores donde un atributo toma sus datos

Relación: Entre atributos.

Tupla: Fila de una relación

Cardinalidad de una relación: Número de tuplas que contiene

Esquema de una relación: Atributos de la relación junto a su dominio

Grado de una relación: Número de atributos en su esquema

Instancia de una relación: Conjunto de tuplas que la componen en cada momento

Como consecuencia de la definición de relación no hay tuplas duplicadas y no existe orden en las filas ni en los atributos.

No se debe permitir que una entidad esté representada en una BD si no se tiene información completa de los atributos que forman las claves de la entidad, es decir, la clave primaria (o parte de ella) no puede tomar valor nulo.

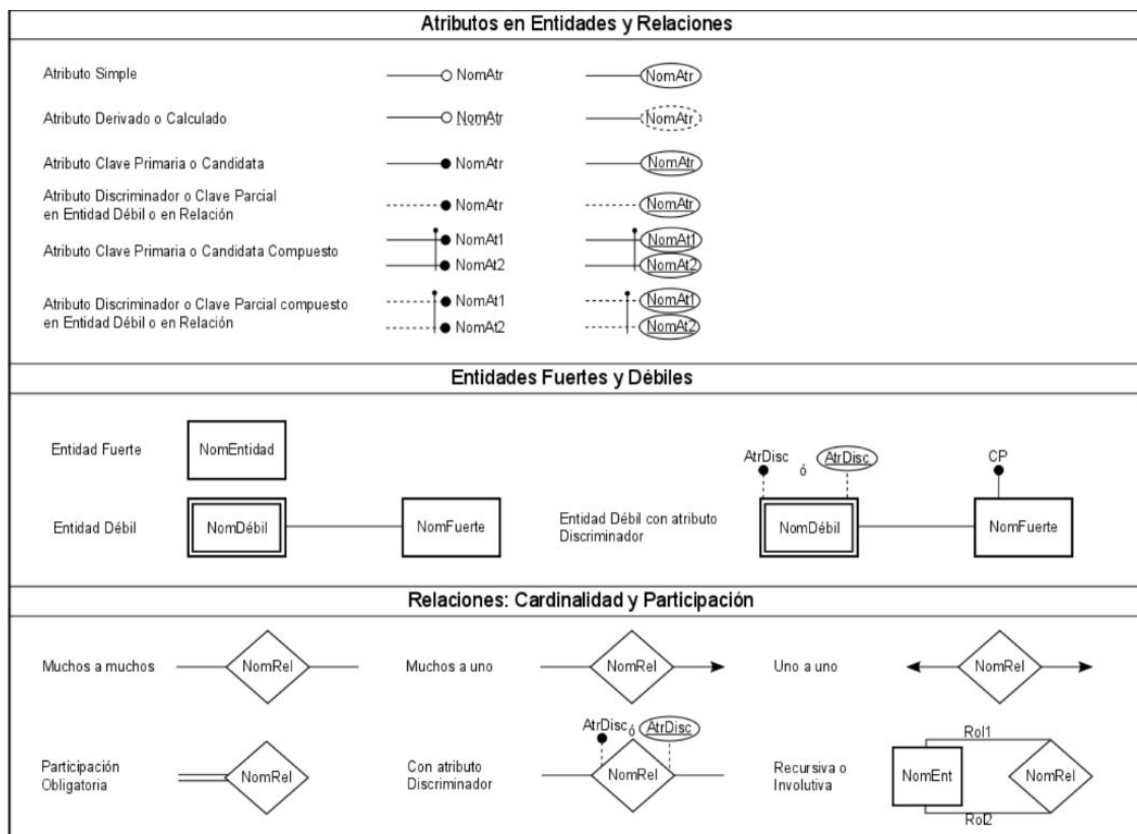
Clave externa: Conjunto de atributos en una relación cuyo contenido debe coincidir con el de la clave primaria en otra relación.

Si una relación incluye una clave externa conectada a una clave primaria, el valor de la clave externa debe ser, bien igual a un valor ya existente en el dominio activo de la clave primaria, o bien completamente nulo (si la semántica lo permite).

El SGBD debe encargarse de cumplir las siguientes restricciones:

- + La unicidad de la clave primaria y de las claves candidatas.
- + La restricción de integridad de identidad: Debe rechazar la inserción y actualización de tuplas con valor NULO en las claves candidatas.
- + Integridad referencial:
 - Rechazar la tupla insertada si el valor de clave externa no concuerda con alguna la clave primaria de una tupla de la relación referenciada
 - Rechazar la tupla insertada también si el valor de clave externa es nulo
 - Si se actualiza la clave externa, rechazar la modificación si se da alguna de las circunstancias mencionadas anteriormente
 - Si se actualiza la clave primaria de la relación referenciada habrá que actualizar en cascada las claves externas que la referencien e impedir la actualización mientras existen referencias al valor anterior.
 - Si se borra la clave primaria de la relación referenciada se hará borrado en cascada de todas las tuplas que la referencian. se pondrá valor nulo en la clave externa de esas tuplas o se impedirá el borrado.

Seminario 2: Diagramas E/R



Seminario 4: Paso a tablas

- Conjunto de entidades fuerte
Asignaturas(Cod_Asig, Nombre, Créditos, Carácter, Curso)
 - Conjunto de entidades débil: La clave primaria está formada por la CP del conjunto de entidades fuerte del que depende y por los atributos discriminadores del propio conjunto. Hay que generar también una clave externa a la tabla del conjunto de entidades fuerte del que depende.
Asignaturas(Cod_Asig, Nombre, Créditos, Carácter, Curso)
Grupos(Cod_asig, Cod_Grupo, Tipo, Max_Al)
 - Relación:
 - Muchos a muchos: La clave primaria está formada por la unión de todos los atributos que son claves primarias
 - Muchos a uno: La clave primaria está formada por la clave primaria de los conjuntos de entidades con cardinalidad muchos.
 - Uno a uno: Dos claves candidatas. Hay que elegir una de ellas como CP y la otra mantenerla como clave candidata.
- En cualquier caso habrá que establecer claves externas a las tablas que referencian.
- Relaciones de Herencia:
 - Para el conjunto general seguir el criterio de conjunto de entidades fuerte.
 - Para las especializaciones se crea una tabla con los atributos propios más la clave primaria del conjunto superior.

Formación
Online
Especializada

Clases Online
Prácticas
Becas

Escuela de
LÍDERES

Jose María Girela
Bim Manager.



La clave primaria es la del conjunto de entidades de nivel superior, además de clave externa al mismo

- Agregaciones: La agregación como tal no se refleja en una tabla específica. Su significado se refleja en la relación que engloba a la agregación.

Fusión de tablas: En ocasiones tras un primer paso a tablas podremos unir algunas para que el conjunto resultante sea el mejor posible. Con ello podremos mejorar la eficiencia siempre que se haga sin pérdida de información (datos, restricciones)

Para la fusión es condición necesaria que tengan la misma CP y que no procedan de herencia, aunque habrá casos en que esto se cumpla y no se pueda hacer la fusión.

Como norma general se podrá fusionar tablas de relaciones 1:n con las entidades de la parte muchos. En relaciones 1:1 con una parte con participación obligatoria se deberá fusionar la tabla de la relación con dicha parte.

Algunas preguntas de exámenes

- El principal objetivo de evitar la redundancia en una BD es ahorrar espacio en disco. (FALSO)
- Cuando se diseña una BD es fundamental conocer las características técnicas del servidor sobre el que se va a implantar. (FALSO)
- El término integridad hace referencia a la veracidad de los datos que se almacenan, esto es, a su correspondencia con la realidad. (VERDADERO)
- La independencia física permite reorganizar las estructuras del nivel interno sin que se vean afectados los programas de aplicaciones. (VERDADERO)
- En el nivel externo se plasma la perspectiva que tiene cada usuario de la BD (VERDADERO)
- UPDATE ... es un comando del DDL. (FALSO)
- Cuando se pasa un diagrama E/R a tablas, las claves candidatas no se tienen en cuenta. (FALSO)
- La independencia física es posible gracias a la correspondencia externa/conceptual. (FALSO)
- Gracias a la transformación conceptual/interna se puede mantener la independencia física. (VERDADERO)
- La elaboración del esquema conceptual es tarea del programador de aplicaciones. (FALSO)
- Una entidad que no tiene clave primaria es siempre una entidad débil. (VERDADERO)
- La forma de implantar la cardinalidad de una relación de un diagrama E/R en una tabla, es mediante la correcta elección de las claves candidatas y primarias. (VERDADERO)
- Una relación de cardinalidad muchos-muchos siempre genera una tabla con clave primaria compuesta. (VERDADERO)
- Una relación de cardinalidad muchos-uno puede generar una tabla con clave primaria compuesta. (VERDADERO)
- CREATE TABLE es un comando del DDL. (VERDADERO)
- En una jerarquía, todas las entidades del conjunto de entidades genérico deben estar en un conjunto de entidades específico. (FALSO)
- En una jerarquía, todas las entidades de un conjunto específico deben estar en el conjunto de entidades genérico. (VERDADERO)
- Todas las tablas procedentes de entidades débiles tienen claves externas. (VERDADERO)
- Un atributo no puede ser clave primaria y externa a la vez. (FALSO)
- La diferencia entre una clave candidata y una primaria es que la candidata no tiene por qué ser minimal. (FALSO)
- Una clave externa puede tomar el valor nulo. (VERDADERO)
- Una clave primaria puede tomar el valor nulo parcialmente. (FALSO)
- La regla de integridad de entidad exige que no existan tuplas duplicadas en una relación. (FALSO)
- La dependencia existencial sólo se da entre entidades débiles y fuertes. (FALSO)
- Todas las restricciones de integridad se pueden mantener eligiendo convenientemente claves candidatas, primarias y externas. (FALSO)