

## Origen y evolución de las bases de datos

==1. Primera generación (años 50)== El fichero no existe, los datos sólo existen dentro de los programas Nuevos soportes: - Cinta perforadora y magnética Acceso secuencial Acceso a ficheros: lenguaje máquina y ensamblador

==2. Segunda generación (años 60)== Diálogo interactivo con la máquina Disco magnético acceso directo (finales 50) Organización secuencial indexada (ISAM) **Sistemas de ficheros:** asociación estática de los ficheros a los programas de forma individual Problemas: - Redundancia - Inconsistencia de datos - Dependencia de datos - Excesivo mantenimiento - Poca flexibilidad frente a cambios - Baja productividad - Limitación de recursos compartidos - Medidas de seguridad

==3. Tercera generación (años 70, pre-relacional)== Distinción entre estructura lógica y física - Arquitectura a dos niveles Unificación de la información sin perder la perspectiva de usuarios Distinción entre significado y valor almacenado

==4. Cuarta generación (años 80, relacional)== Nuevos productos: Sistemas de Bases de Datos Control centralizado evita redundancia Clara distinción entre el modelo lógico y el físico Alto grado de independencia de datos Almacenamiento transparente al usuario Lenguajes más potentes (Qué en lugar de Cómo) Modelo relacional CODD: - Basado en el álgebra y la teoría de conjuntos - Prototipos - INGRES Universal de Berkeley (1973-75) - SYSTEM-R de IBM (1974-77) - Sistemas comerciales - INGRESS DE RTI (1980) - SQL/DS de IBM (1981) - ORACLE de RSI (1981) - DB2 de IBM(1983)

==5. Quinta generación (años 90, post-relacional)== Bases de datos deductivas Bases de datos orientadas a objetos Bases de datos activas

## Concepto de base de datos: objetivos de las bases de datos

### 1. Objetivos

==Independencia de datos (Flexibilidad)== - Los cambios en las aplicaciones no deben imponer un nuevo diseño y viceversa ==Coste mínimo== - Adaptación rápida y con coste mínimo a las nuevas características de la empresa ==Consistencia y mínima redundancia (Integridad)== - Evitar repetición innecesaria de datos y la incoherencia entre estos ==Datos compartidos== - Permite una mayor disponibilidad de los datos ==Versatilidad en la representación de los datos== - Distintos usuarios quieren ver de forma distinta los datos ==Capacidad de búsqueda== - Rapidez en la exploración de la Base de Datos ==Integridad== - Garantizar la exactitud y veracidad de los datos ==Tolerancia a fallos y seguridad== - Protección de los datos ante accesos no autorizados

### 2. Base de Datos:

Colección de datos estructurados según un modelo que refleje las entidades, relaciones y restricciones existentes en el mundo real Los datos han de ser compartidos por diferentes usuarios y aplicaciones, deben mantenerse independientes de éstas, y su definición y su descripción deben ser únicas, estando almacenadas junto a los mismos Los tratamientos habrán de conservar la integridad y seguridad de los datos

### 3. Transacción

Conjunto de operaciones que deben realizarse de forma atómica (todas o ninguna) Garantizar la consistencia de los datos Debe estar aislada de otras transacciones Debe perdurar en el tiempo, los cambios que produce deben quedar reflejados en la base de datos SQL (Structure Query Language) - COMMIT/ROLLBACK

## Independencia de datos. Arquitectura de bases de datos

### 1. Implica la separación lógica y física de la base de datos

Influencian en la arquitectura del sistema Grado de dependencia Fase o etapa del proceso donde se efectúa la transformación entre niveles

### 2. La independencia de datos es (i):

La capacidad de un sistema de gestión de bases de datos para permitir que las referencias a los datos almacenados, especialmente en los programas y sus descripciones de los datos, estén aisladas de los cambios y de los diferentes usos en el entorno de los datos, como pueden ser: - La forma de almacenar dichos datos - El modo de compartirlos con otros programas - Cómo se reorganizan para mejorar el rendimiento del sistema de bases de datos

#### 2.1 La independencia de datos es (ii)

La inmunidad de las aplicaciones ante cambios de la estructura de almacenamiento y de los métodos de acceso

### 3. Tipos de dependencia

De descripción: - Separación de la definición de datos a nivel físico lógico De manipulación: - Independencia respecto a los cambios de acceso y soportes físicos

### 4. Inmunidad del usuario frente a:

Atributos: Nombre, tamaño, tipo... Entidades: Nombre, nuevos atributos... Estructuras: Cambio en nombre de relaciones, cardinalidad... Nivel intermedio: Tamaño de bloques, longitud de registros, accesos... Nivel físico: Tipo de soporte, tamaño de los volúmenes, sistema operativo...

### 5. Arquitectura a tres niveles:

Propuesta por el comité ANSI/X3/SPARC Asocia a cada nivel un esquema:

- Esquema externo
  - o Comprende las vistas individuales de los usuarios (subesquemas)
  - o Extracto de la vista conceptual
- Esquema conceptual
  - o Vista lógica general de todos los datos
  - o Interrelaciones entre los datos
  - o Uso de un modelo de datos que proporcione un nivel de abstracción de la vista interna
  - o Uso de un lenguaje de definición de datos (DDL)
- Esquema interno
  - o Es el nivel más próximo al almacenamiento físico
  - o Trata los datos como registro internos (no como bloques de datos)
  - o Contiene información sobre:
    - ☐ Estructuras de datos usadas

- ☐ Mecanismo de acceso
- ☐ Distribución de registros en el espacio lógico

![[Pasted image 20250918125052.png]]

## 6. Interfaces entre niveles:

Los proporciona el SGBD (Sistema Gestor de Base de Datos) Transformaciones entre los esquemas Consultas y actualizaciones: DML (Lenguaje de Manipulación de Datos): - Lenguaje interactivo - Precompilador - Extensión del compilador Creación de las estructuras: DDL (Lenguaje de Definición de Datos): - DDL interno - DDL conceptual - DDL externo

## Sistemas de gestión de bases de datos (SGBD)

Conjunto de programas que actúan como intermediario entre los datos y los usuarios Recoge la petición del usuario y responde a ellas ![[Pasted image 20250924114522.png]]

### Características:

Descripción de la BBDD exterior a los programas y gestionada por el SGBD Los programas no leen ni graban directamente sobre la estructura interna de la BBDD Gestión de: - Control de accesos concurrentes - Control de autorizaciones de acceso Reconstrucción y/o restauración de la BBDD

### Diccionario de datos:

Es una BBDD que contiene información sobre la propia BBDD (metadatos) Debe suministrar documentación única y actualizada de forma que pueda ser usada para obtener información sobre los tipos de datos almacenados y cómo se deben usar.

### Componentes de un SGBD:

-Procesador de I/O: Toma órdenes y muestra la respuesta -Analizador: Análisis sintáctico de la orden -Control de autorizaciones: Chequeo de autorizaciones Diccionario y obtención de código intermedio -Procesador de actualizaciones y control de integridad: ejecución de la actualización y control de asistencia -Procesador de consultas y optimizador: Transforma la consulta en términos conceptuales y reformulación para la optimización del acceso -Generador de código ejecutable: Secuencia de lecturas y escrituras en disco -Gestos de transacciones: Sincronización de los accesos concurrentes - Gestor de recuperación: Recuperación del estado de la BBDD previo al fallo -Gestos de datos: Gestión de hardware y ejecución de los accesos físicos

### Estructura de SGBD por interfaces:

![[Pasted image 20250924115603.png]] **Arquitectura cliente-servidor:** El SGBD actúa como servidor proporcionando todo el soporte de los niveles externo, conceptual, e interno Las aplicaciones que se ejecutan sobre SGBD actúan como cliente y las interfaces de consulta que interactúan con el usuario y envían consultas u otros comandos al servidor Ej: los sistemas relacionales utilizan el lenguaje SQL para respresentar peticiones del cliente ante el servidor. El servidor las procesa y envía la respuesta al cliente en forma de tabla o relación

### Estructura de un SGBD multiusuario:

Arquitectura cliente-servidor (funciones del servidor): Aceptar y procesar las peticiones de los clientes de la BBDD Comprobar autorizaciones Asegurar que se cumplen las restricciones de integridad Realizar los procesos de consulta/actualización y transmitir la respuesta al cliente Mantener el diccionario de datos Proporcionar acceso concurrente a la BBDD...

## Administración de bases de datos

### Tareas de administración:

La estructura de la base de datos Descripción conceptual: Una vez conocidos los requisitos de información, es preciso realizar el diseño conceptual Descripción física de los datos: Encontrar una estructura interna que soporte el esquema lógico y los objetivos de diseño con la máxima eficiencia de los recursos Especificaciones de vistas o subesquemas Los estándares: En cuento a documentación, metodologías Procedimientos de explotación y uso Aspectos relativos a seguridad, integridad y confidencialidad

### Herramientas que utiliza el administrador:

Lenguajes de definición de datos (DDL) Utilidades del SGBD (copias de seguridad, asignación de usuarios, simulación de procesos, documentación...) Diccionario de datos: Contendrá la definición y descripción de todos los datos (metadatos). Ayuda a la gestión de la información como recurso y conseguir la integración de la semántica de forma centralizada

### Tipos de administración

1. El administrador a nivel de empresa (Nivel Conceptual)
2. El administrador de la base de datos (Nivel Interno)
3. Los administradores por aplicaciones (Nivel Externo)