

Clase 1

Base de datos

- ¿ que es ?
 - Es una colección de datos relacionada y de archivos que sirven a diferentes aplicaciones
 - Un dato representa hechos conocidos que pueden registrarse y tienen un resultado implícito.
- Propiedades
 - Representa aspectos del mundo real, a veces denominado “universo de discurso”
 - Una colección de datos random no puede llamarse base de datos, deben tener cierta lógica y relación
 - Esta sustentada en dispositivos de almacenamiento persistente

Archivos

- Son
 - Colección de registros guardados en un almacenamiento secundario
 - Colección de datos almacenados en dispositivos de almacenamiento secundario
 - Colección de registros que abarcan entidades con un aspecto común y originados para un propósito particular
- Organización
 - Secuencia de bytes
 - No se puede determinar fácilmente el comienzo y el fin de cada dato
 - Se da en archivos de texto
 - Registros y campos
 - campo es la unidad mas pequeña significativa de un archivo
 - registro es un conjunto de campos agrupados que definen un elemento del archivo
 - acceso
 - **secuencial físico:** acceso a los registros uno tras otro en el orden en que están guardados
 - **secuencial indizado:** acceso a los registros según un orden definido en otra estructura
 - **directo:** se accede a un registro determinado sin necesidad de pasar por las anteriores
 - según su forma de acceso
 - **serie:** en archivos secuenciales físicos, para acceder a un registro hay que pasar si o si por los anteriores
 - **secuencial:** se acceden según el orden de alguna clave, en secuencial indizado
 - **directo:** se accede al dato deseado (directo)

- buffers
 - memoria intermedia entre un archivo y un programa, donde los datos están provisoriamente antes de ser almacenados en la memoria secundaria o donde están después de ser leídos de la memoria
 - ocupan lugar en la ram
 - el sistema operativo los maneja
-

Clase 2 – no es necesaria

Clase 3

Viaje de un byte

- Quienes se involucran
 - Administrador de archivos
 - Conjunto de programas del sistema operativo que tratan aspectos relacionados con archivos y dispositivos E/S
 - En capas superiores
 - Aspectos lógicos de datos (tabla)
 - Establecer si las características del archivo permiten o son compatible con la operación deseada
 - En capas inferiores
 - Aspectos físicos (FAT)
 - Determinar donde se guarda el dato (cilindro, superficie, sector)
 - Si el sector ya se encuentra en ram se usa, de lo contrario se trae a ram previamente
 - Buffer de datos
 - Agilizan los movimientos de E/S
 - Implican trabajar con grandes grupos de datos en ram para que se acceda lo menos posible a la memoria secundaria
 - Procesador E/S
 - Es un dispositivo que se encarga de la transmisión desde o hacia almacenamientos externos independientes de la cpu
 - Controlador del disco
 - Controla el funcionamiento del disco
 - Colocarse en la pista
 - Colocarse en el sector
 - Realizar la transferencia
- Capas en el protocolo de transmisión de un byte
 1. El programa pide al S.O escribir el contenido de una variable en un archivo
 2. El S.O transmite el trabajo al administrador de archivos o **adm**

3. El adm busca al archivo en la tabla de archivo y verifica las características
4. El adm obtiene de la fat la direccino física del sector donde se va a guardar el byte
5. El adm se aseguro que el sector este en el buffer (ram) o si no lo trae, graba el byte donde va en el sector, en el buffer
6. El adm le da las instrucciones al procesador de E/S de donde esta el byte en la ram y en que partes del disco tiene que ir
7. El procesador E/S envia el dato al controlador del disco con la dirección de escritura
8. El controlador del disco transfiere bit a bit el byte al disco

Tipo de archivos

- Tipo de archivos
 - Secuencia de bytes
 - No se puede determinar fácilmente el comienzo y el fin de un dato
 - Ej: los archivos de texto
 - Archivos estructurados
 - Registros de longitud fija o variable
 - Campos de longitud fija o variable
 - Unidad significativa mas pequeña para almacenar información
 - Identidad de campos
 - Longitud predecible (longitud fija), se desperdicia espacio si el tamaño del dato no ocupa todo el campo
 - Indicador de longitud al principio del campo
 - Separador de campos al final de cada campo
 - Estáticos
 - Sufren pocos cambios
 - Pueden actualizarse por lotes
 - No necesita de estructuras adicionales para agilizar las operaciones
 - Volátiles
 - Tienen muchas modificaciones
 - Se deben organizar para que las operaciones se realicen mas rápido
 - Se suelen usar estructuras adicionales

Claves

- Tienen como objetivo agarrar un registro en concreto en lugar de todo el archivo
- Es conveniente identificar un registro con una clave que se relacione con su contenido
- Tipos
 - Unicova o primaria, identificar un registro en concreto dentro del archivo, no se puede repetir esta clave
 - Secundaria, identifica un grupo de registros
- Forma canonica, es una forma estándar para ingresar todas las claves, todo en mayúsculas, todo en minúsculas, etc
- Ordenar las claves (siendo n registros)
 - Acceso secuencial
 - En promedio son $n/2$ comparaciones siendo $O(n)$

- Leyendo de a bloques de registros mejora la cantidad de accesos a disco pero no las comparaciones
- Acceso directo
 - Se necesita conocer el lugar donde esta el registros
 - Tiene $O(1)$

Operaciones

- Bajas
 - Cualquier estrategia de eliminación debe proveer una forma de reconocerlos después de borrarlos, ej una marca de borrado
 - Se puede cancelar el borrado fácilmente
 - Baja física
 - Baja lógica
 - Compactación
 - Recupera espacio
 - Copiar todo en un archivo nuevo menos los eliminados (baja física)
 - Aprovechar el espacio
 - Registros de longitud fija
 - Necesidad de marcas de borrado (baja logica)
 - Registros de longitud variable
 - Los elementos nuevos deben “caber” dentro del espacio del elemento eliminado
 - Recuperación del espacio
 - Registros de longitud fija
 - Recuperación de espacio con lista o pilas (headers)
 - Lista encadenada de registros disponibles
 - Al insertar un registros nuevo cualquier borrado sirve por que son de longitud fija
 - La lista no necesita tener un orden
 - Registros de longitud variable
 - El problema es que no puede ir en cualquier lugar por el tamaño
 - Marcas de borrado igual que en longitud fija
 - Como el nrr solo se puede usar en reg long fija, como enlace se utiliza un campo binario que explícitamente indica el enlace
 - Cada registro indica al inicio su cantidad de byes
 - Funcionamiento
 - Se busca el primer registro borrado que sirva (con suficiente tamaño)
 - Como hay que buscarlo no se puede organizar la lista como una pila
 - Genera fragmentación
 - Estrategias de colocación
 - Primer ajuste
 - Se elige el primer registro borrado útil
 - Minimiza la busqueda

- No se preocupa por la fragmentación
 - Mas rapido
- mejor ajuste
 - elige el registro borrado de tamaño mas parecido al nuevo
 - la búsqueda es mas compleja
 - genera fragmentación interna
- Peor ajuste
 - Elige el registro borrado mas grande y se le asigna solo lo necesario
 - El resto se deja para un nuevo registro
 - Genera fragmentación externa
- Fragmentación
 - Interna
 - Ocurre en registro de longitud fija
 - Cuando un registro va en el lugar de uno borrado pero no ocupa todo el espacio
 - Externa
 - En registros de longitud variable
 - En general es mucho menor a la fija, ya que el espacio que sobra al sobrescribir un registro si este no ocupa todo el espacio que tenia el anterior, se utiliza para otro registro
 - Si el espacio es demasiado chico se le llama residuo o fragmentación externa
 - Aprovechar el espacio fragmentado
 - Externa
 - Unir espacios libres para hacer uno mayor que sirva
 - Minimizar la fragmentación usando el registro borrado mas adecuado para el nuevo registro

Clase 4

Búsquedas

- Búsqueda secuencial
 - Pasa por todos los registros
- Búsqueda directa
 - Necesita el nrr
- Búsqueda binaria
 - Necesita el archivo ordenado por claves
 - Tiene que ser de longitud fija para poder tener el nrr
 - No es especialmente útil por que para archivos muy grandes que no entren en ram no sirve
 - Se pueden llevar solo las claves a ram pero puede pasar que tampoco entren
 - Se podría partir el archivo, ordenar cada parte y luego hacer un merge

Clase 5

Índices

- Es una herramienta para encontrar registros en un archivo, consiste en que cada registro tenga un campo de clave y un campo de referencia que indique donde encontrar el registro dentro del archivo
- Funciona como una tabla que contiene las claves de los registros y asociada a cada clave información que sirve para encontrar rápidamente el registro (si los registros son de longitud fija esta información sería el nrr y si son de longitud variable sería la distancia en bytes)
- Permite ver un archivo desordenado como uno ordenado según el orden de las claves
- El archivo de índices sería de longitud fija, ordenado para poder hacer una búsqueda binaria en ram
- Actualización de registros
 - Si no se cambia la clave
 - Si el registro no cambia de longitud, no se toca
 - Si cambia de longitud y se reubica en el archivo de datos se debe guardar la nueva posición inicial en el índice
 - Si es de longitud fija no hay que hacer nada
 - Si se modifica la clave
 - Se debe actualizar el archivo de índices
- Si el archivo de índices no entra en ram
 - Uso de árboles
 - Hashing
- Si es necesario buscar en el archivo de índices por una clave secundaria
 - Se usa otro archivo de índices secundarios que relacione los índices secundarios con los primarios
 - Trae problemas ya que se repite información y se debe reacomodar con cada adición
 - Una solución es por cada clave secundaria cargar un arreglo con las claves primarias asociadas a esa secundaria
 - Este a su vez trae algunos problemas como
 - fragmentación interna
 - espacio insuficiente
 - mejora
 - por cada clave en lugar de un arreglo usar una lista de punteros
 - problema, podría haber muchos desplazamientos en disco lo que sería costoso si hay muchos índices