

Sistema de Ficheros

Sistemas Operativos - ITIG

Álvaro Polo Valdenebro

apoloval@gsync.es

Abril 2009



©2009 GSyC
Algunos derechos reservados.
Este trabajo se distribuye bajo la licencia
Creative Commons Attribution Share-Alike
disponible en <http://creativecommons.org/licenses/by-sa/2.1/es>



- 1 Interfaz del sistema de ficheros
- 2 Implementación

Contenidos

- 1 Interfaz del sistema de ficheros
- 2 Implementación

Objetivos

Queremos guardar información

- De gran tamaño
- De carácter persistente
- Que se pueda compartir
- Con garantías de integridad y seguridad

Los discos y otros dispositivos de almacenamiento nos lo dan,
pero...

Objetivos

Queremos guardar información

- De gran tamaño
- De carácter persistente
- Que se pueda compartir
- Con garantías de integridad y seguridad

Los discos y otros dispositivos de almacenamiento nos lo dan, pero...

- Son heterogéneos
- Son complejos de usar
- Queremos abstracciones lógicas

Fichero o archivo

- Unidad lógica de almacenamiento
- Colección de información relacionada
- Se aloja en memoria persistente (disco)
- Discusiones de diseño
 - **Tipo**, naturaleza de la información que contiene
 - **Estructura**, cómo se distribuye la información
 - **Métodos de acceso**, cómo accedemos a la información
 - **Nombrado**, cómo se referencian
 - **Permisos y protección**, cómo se evita el uso no autorizado
 - **Implementación**, cómo se logra

Atributos o metadatos

- **Nombre**, usado por los humanos para referenciar el archivo
- **Identificador**, usado por la máquina para referenciar el archivo
- **Tipo**, indica (en los sistemas que lo soporten) qué tipo de información contiene
- **Ubicación**, indica dónde está almacenado en el disco
- **Tamaño**, indica cuanta información almacena
- **Propietario y permisos**, indican quien puede acceder a la información
- **Fecha de creación, modificación, acceso**, nos informa cuándo ha sido manipulada la información

Operaciones

- **Creación**, genera un nuevo archivo en el sistema
- **Escritura**, añade datos a un archivo a partir del puntero de escritura
- **Lectura**, obtiene datos de un archivo a partir del puntero de lectura
- **Reposicionamiento**, cambia la posición del puntero de lectura/escritura
- **Borrado**, elimina el fichero del sistema
- **Truncado**, elimina la información anterior y escribe sobre ella
- **Consulta de metadatos**, obtiene los atributos
- **Cambio de metadatos**, modifica los atributos

Tipos de fichero

- Es útil conocer la naturaleza de la información de un fichero
 - ¿Es OpenDocument? ¿Es MP3? ¿Es JPEG? ¿Es HTML? ¿Es C?
- Se evitan confusiones al conocer las restricciones de cada tipo
 - ¿Ejecutar un JPG? ¿Imprimir un programa ejecutable?
- Tipos mediante extensión
 - El nombre se complementa con un código que indica el tipo
 - .exe, .bat, .com, .jpg, .c
 - Puede ser estricta o no estricta
 - Si cambiamos la extensión, podemos “engañar” al programa
- Tipos mediante números mágicos
 - Al comienzo, se incluye una pequeña secuencia de bytes
 - Esta secuencia identifica el tipo de fichero
 - No es tan fácil engañar al programa

Estructura

Determina cómo se almacena la información en el fichero

- Estructura interna de los datos
 - Fichero binario → segmento de datos, segmento de código, ...
 - Fichero de texto → carácter 1, carácter 2, ...
- Si imponemos un conjunto de estructuras...

Estructura

Determina cómo se almacena la información en el fichero

- Estructura interna de los datos
 - Fichero binario → segmento de datos, segmento de código, ...
 - Fichero de texto → carácter 1, carácter 2, ...
- Si imponemos un conjunto de estructuras...
 - Poca flexibilidad → ¿y si el sistema no soporta mi formato?
 - El SO debe implementar los algoritmos de lectura/escritura
- Si no lo imponemos...

Estructura

Determina cómo se almacena la información en el fichero

- Estructura interna de los datos
 - Fichero binario → segmento de datos, segmento de código, ...
 - Fichero de texto → carácter 1, carácter 2, ...
- Si imponemos un conjunto de estructuras...
 - Poca flexibilidad → ¿y si el sistema no soporta mi formato?
 - El SO debe implementar los algoritmos de lectura/escritura
- Si no lo imponemos...
 - Mayor flexibilidad → admitimos cualquier estructura
 - El programa debe implementar la lectura/escritura

Métodos de acceso

Cómo se accede a la información

- **Acceso secuencial**, los datos se leen uno tras otro
 - Partimos del primer elemento
 - Al leer/escribir, apuntamos al siguiente elemento
 - Podemos reiniciar el puntero de datos (o situarlo en una posición concreta)
 - Inspirado en...
- **Acceso directo**, podemos acceder a cualquier elemento del fichero
 - Proporcionamos la posición del elemento cuando queremos leer/escribir
 - Inspirado en...

Métodos de acceso

Cómo se accede a la información

- **Acceso secuencial**, los datos se leen uno tras otro
 - Partimos del primer elemento
 - Al leer/escribir, apuntamos al siguiente elemento
 - Podemos reiniciar el puntero de datos (o situarlo en una posición concreta)
 - Inspirado en... cintas magnéticas
- **Acceso directo**, podemos acceder a cualquier elemento del fichero
 - Proporcionamos la posición del elemento cuando queremos leer/escribir
 - Inspirado en...

Métodos de acceso

Cómo se accede a la información

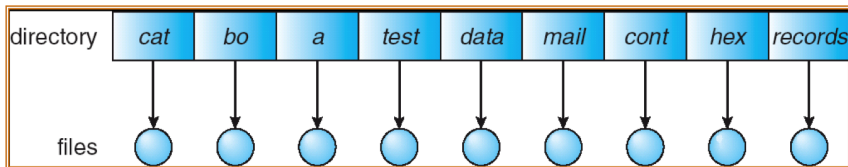
- **Acceso secuencial**, los datos se leen uno tras otro
 - Partimos del primer elemento
 - Al leer/escribir, apuntamos al siguiente elemento
 - Podemos reiniciar el puntero de datos (o situarlo en una posición concreta)
 - Inspirado en... cintas magnéticas
- **Acceso directo**, podemos acceder a cualquier elemento del fichero
 - Proporcionamos la posición del elemento cuando queremos leer/escribir
 - Inspirado en... discos de acceso aleatorio

Directorios

- Los ficheros se almacenan en volúmenes físicos
- Para gestionarlos, se guarda una relación de ellos en un **directorio**
 - Nombre del fichero
 - Ubicación
 - Tamaño
 - Tipo
 - ...
- Realizamos operaciones sobre el directorio
 - Buscar archivos
 - Crear archivos
 - Borrar archivos
 - Listar el directorio
 - ...

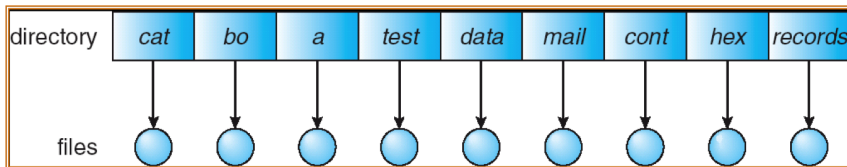
Directorio de un nivel

- Un único directorio almacena todos los ficheros
- Problema...



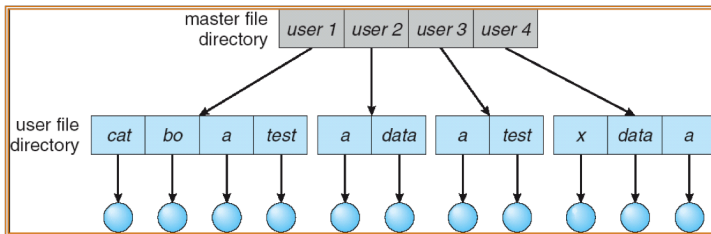
Directorio de un nivel

- Un único directorio almacena todos los ficheros
- Problema...
 - Los nombres no pueden repetirse
 - Varios usuarios pueden querer emplear el mismo nombre
 - La gestión de los ficheros para el usuario podría ser demasiado compleja



Directorio de dos niveles

- El **directorio de archivos de usuario** (UFD) mantiene los ficheros de cada usuario
- El **directorio maestro de ficheros** (MFD) mantiene entradas a los UFD.
- Cada usuario opera sobre su propio UFD



Directorio de dos niveles: consideraciones

Se resuelve la colisión de nombres, pero

- ¿Cómo compartimos ficheros con otros usuarios?...

Directorio de dos niveles: consideraciones

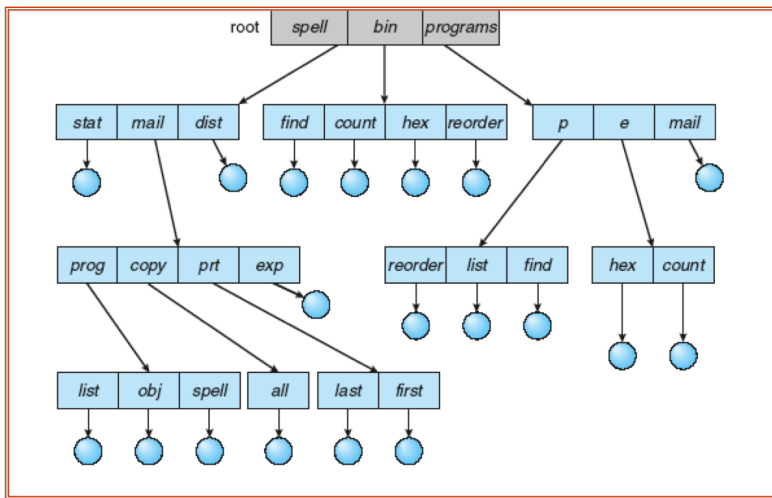
Se resuelve la colisión de nombres, pero

- ¿Cómo compartimos ficheros con otros usuarios?...
- Podemos permitir el acceso a ficheros de otros UFDs
- Concepto de **nombre de ruta**
 - Necesitamos referenciar ficheros en otros UFDs
 - El nombre data se busca en mi UFD
 - El nombre /fernando/data se busca en el UFD de Fernando
 - Otras variantes: C:\Fernando\data, ...
- Concepto de **ruta de búsqueda**
 - Los ficheros de sistema deberían poder buscarse fuera del UFD
 - Probablemente ls no esté en mi UFD, y no quiero teclear /system/ls constantemente
 - Defino una ruta de búsqueda en la cual localizar los ficheros

Arbol de directorios

- Generalización del directorio de dos niveles a una altura arbitraria
- Los directorios pueden contener ficheros o subdirectorios
- El subdirectorio es un archivo “especial”
 - Un bit determina si es archivo común o subdirectorio
- Cada proceso tiene asociado un **directorio actual**
 - Aloja la mayor parte de los ficheros que usará el proceso
 - Es el primer elemento de la ruta de búsqueda
 - Podemos cambiarlo con una llamada al sistema
- Ahora distinguimos dos tipos de nombres de ruta
 - **Absolutos** cuando referencian a partir del directorio raíz
 - **Relativos** cuando referencian a partir del directorio actual

Árbol de directorios: ejemplo



Directorios en grafo acíclico

¿Cómo podemos compartir un fichero o subdirectorio manteniéndolo en nuestro propio directorio?...

Directorios en grafo acíclico

¿Cómo podemos compartir un fichero o subdirectorio manteniéndolo en nuestro propio directorio?...

- Generalizamos el árbol de directorios en un grafo acíclico
- Permitimos que un fichero o subdirectorio pertenezca a varios directorios
- Nuevo tipo de entrada de directorio: **enlace**
 - Simbólico → almacena una referencia en forma de ruta
Al borrar...
 - Fuerte → distintos nombres para los mismos datos
Al borrar...

Directorios en grafo acíclico

¿Cómo podemos compartir un fichero o subdirectorio manteniéndolo en nuestro propio directorio?...

- Generalizamos el árbol de directorios en un grafo acíclico
- Permitimos que un fichero o subdirectorio pertenezca a varios directorios
- Nuevo tipo de entrada de directorio: **enlace**
 - Simbólico → almacena una referencia en forma de ruta
Al borrar... dejamos los enlaces “colgando”
 - Fuerte → distintos nombres para los mismos datos
Al borrar...

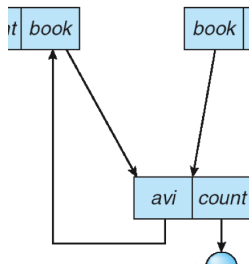
Directorios en grafo acíclico

¿Cómo podemos compartir un fichero o subdirectorio manteniéndolo en nuestro propio directorio?...

- Generalizamos el árbol de directorios en un grafo acíclico
- Permitimos que un fichero o subdirectorio pertenezca a varios directorios
- Nuevo tipo de entrada de directorio: **enlace**
 - Simbólico → almacena una referencia en forma de ruta
Al borrar... dejamos los enlaces “colgando”
 - Fuerte → distintos nombres para los mismos datos
Al borrar... contadores de referencias

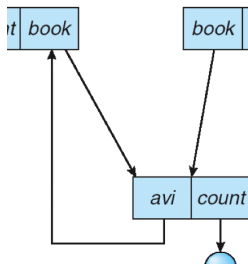
Directorios en grafo general

- Los enlaces pueden generar ciclos
- Problemas...



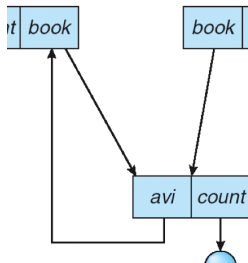
Directorios en grafo general

- Los enlaces pueden generar ciclos
- Problemas...
 - Recorrer el árbol sin bucles cíclicos
 - Borrado de enlaces duros
- Soluciones al borrado...



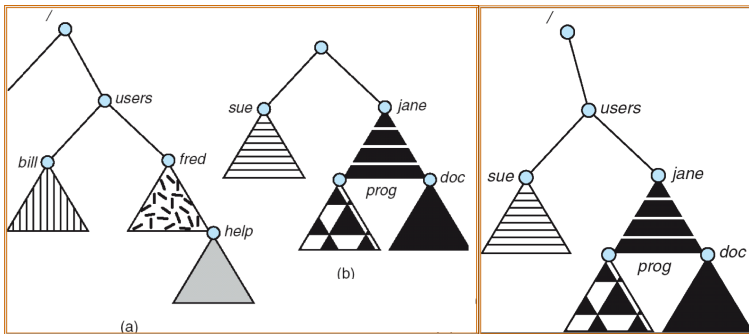
Directorios en grafo general

- Los enlaces pueden generar ciclos
- Problemas...
 - Recorrer el árbol sin bucles cíclicos
 - Borrado de enlaces duros
- Soluciones al borrado...
 - Recolector de memoria
 - Cálculo de ciclos
 - Restricción de ciclos



Montaje de sistemas de archivos

- Los sistemas de ficheros se **montan**



- Se escoge un **punto de montaje** donde ubicar el sistema de ficheros

Permisos y protección

Debemos proteger el acceso no autorizado a los ficheros

- Cada fichero tiene asociado un usuario propietario y un grupo
- Tipos de acceso
 - **Lectura**
 - **Escritura**
 - **Ejecución**
 - Otros: adición, borrado, listado, ...
- Listas de control de acceso (ACL)
 - Problema...

Permisos y protección

Debemos proteger el acceso no autorizado a los ficheros

- Cada fichero tiene asociado un usuario propietario y un grupo
- Tipos de acceso
 - **Lectura**
 - **Escritura**
 - **Ejecución**
 - Otros: adición, borrado, listado, ...
- Listas de control de acceso (ACL)
 - Problema... muy largas
 - Solución...

Permisos y protección

Debemos proteger el acceso no autorizado a los ficheros

- Cada fichero tiene asociado un usuario propietario y un grupo
- Tipos de acceso
 - **Lectura**
 - **Escritura**
 - **Ejecución**
 - Otros: adición, borrado, listado, ...
- Listas de control de acceso (ACL)
 - Problema... muy largas
 - Solución... ACL simple → propietario, grupo y otros

Contenidos

- 1 Interfaz del sistema de ficheros
- 2 Implementación**

- **Bloque de control de arranque:** primer bloque del disco, contiene el cargador del SO
- **Bloque de control de fichero (FCB)**
 - Metadatos de fichero
 - Relación de bloques
 - Unix: inodo
- **Bloque de control de volúmen**
 - Número y tamaño de los bloques
 - Relación de bloques libres
 - Relación de FCBs
 - Unix: superbloque
- **Estructura de directorios:** relación de nombres-FCBs

Directorios

Disposición

- Tratados a menudo como ficheros
- El contenido es una lista de tuplas “nombre de fichero - FCB”

Organización interna

- Lista lineal
 - Lista enlazada de tuplas
 - Problema...

Directorios

Disposición

- Tratados a menudo como ficheros
- El contenido es una lista de tuplas “nombre de fichero - FCB”

Organización interna

- Lista lineal
 - Lista enlazada de tuplas
 - Problema... complejidad en la búsqueda/inserción
- Tabla hash
 - Problema...

Directorios

Disposición

- Tratados a menudo como ficheros
- El contenido es una lista de tuplas “nombre de fichero - FCB”

Organización interna

- Lista lineal
 - Lista enlazada de tuplas
 - Problema... complejidad en la búsqueda/inserción
- Tabla hash
 - Problema... tamaño dinámico de tabla

Asignación

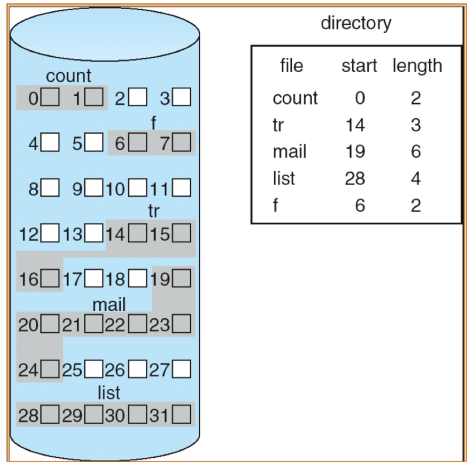
- Los ficheros se almacenan en disco
- Cada fichero usará uno o varios bloques de disco
- Inevitablemente tendremos...

Asignación

- Los ficheros se almacenan en disco
- Cada fichero usará uno o varios bloques de disco
- Inevitablemente tendremos... fragmentación interna
- Debemos decidir cómo asignar los bloques a los ficheros
 - Asignación continua
 - Asignación enlazada (DOS/Windows 9x FAT)
 - Asignación indexada (Unix)

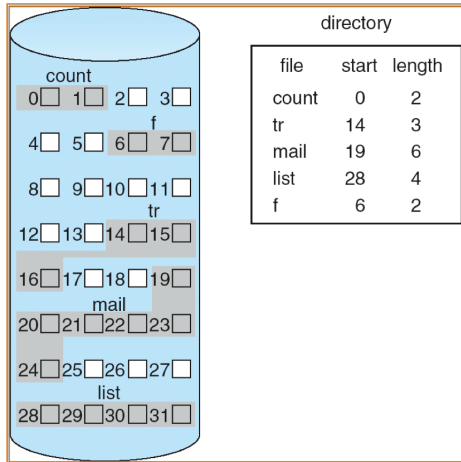
Asignación continua

- Se asigna un conjunto de bloques contiguos
- El FCB/directorio indica el primer bloque y el número de bloques
- Ventajas...
- Problemas...



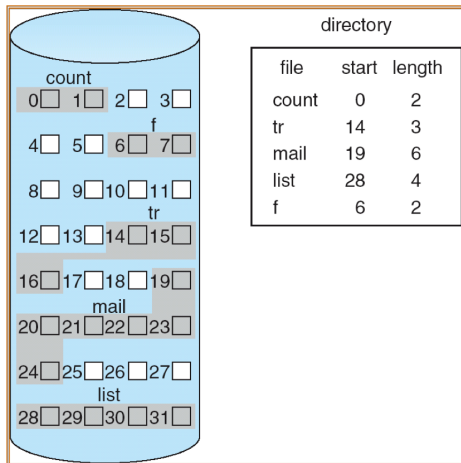
Asignación continua

- Se asigna un conjunto de bloques contiguos
- El FCB/directorio indica el primer bloque y el número de bloques
- Ventajas... óptimo en lecturas (especialmente directas)
- Problemas...



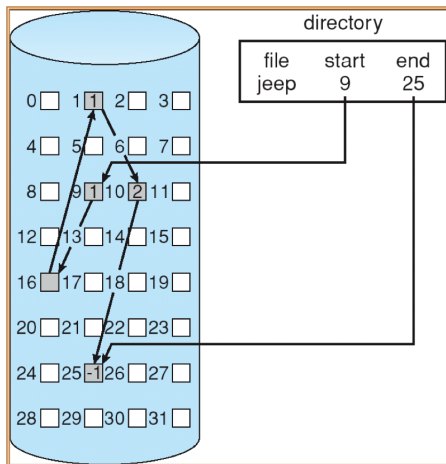
Asignación continua

- Se asigna un conjunto de bloques contiguos
- El FCB/directorio indica el primer bloque y el número de bloques
- Ventajas... óptimo en lecturas (especialmente directas)
- Problemas...
 - Fragmentación externa
 - Crecimiento de ficheros



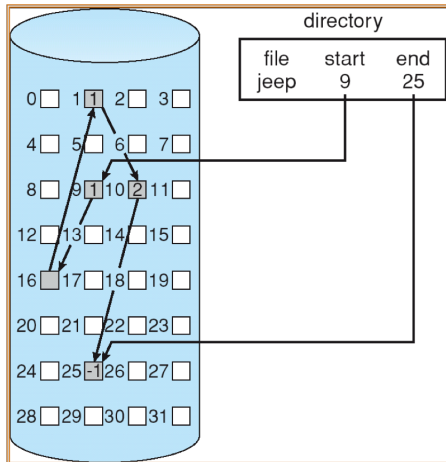
Asignación enlazada

- Los bloques contienen punteros al próximo bloque dentro del fichero
- El FCB/directorio indica el primer bloque (y quizás el último)
- Ventajas...
- Problemas...



Asignación enlazada

- Los bloques contienen punteros al próximo bloque dentro del fichero
- El FCB/directorio indica el primer bloque (y quizás el último)
- Ventajas... sin fragmentación externa
- Problemas...



Asignación enlazada

- Los bloques contienen punteros al próximo bloque dentro del fichero
- El FCB/directorio indica el primer bloque (y quizás el último)
- Ventajas... sin fragmentación externa
- Problemas...
 - Poco óptimo en accesos directos
 - Los punteros desperdician espacio → usamos *clusters*

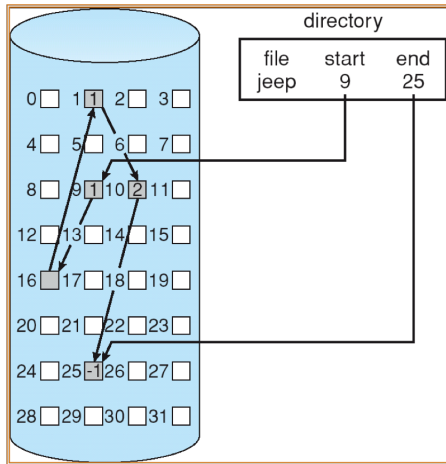
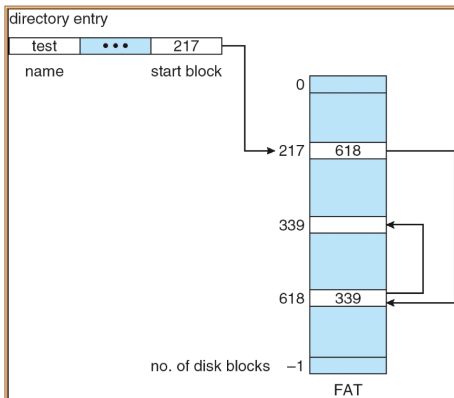


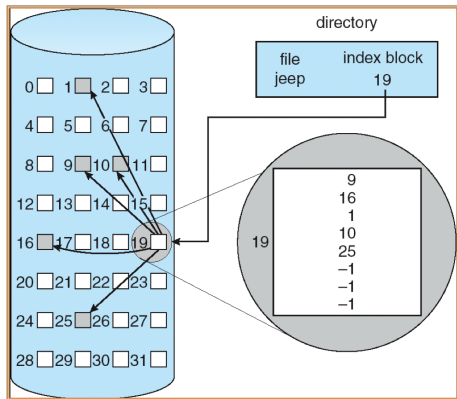
Tabla de asignación de archivos (FAT)



- Variante de asignación enlazada en DOS/Windows 9x
- Emplea una tabla que enlaza los bloques de cada fichero
- El FCB/directorio indica el primero bloque del archivo

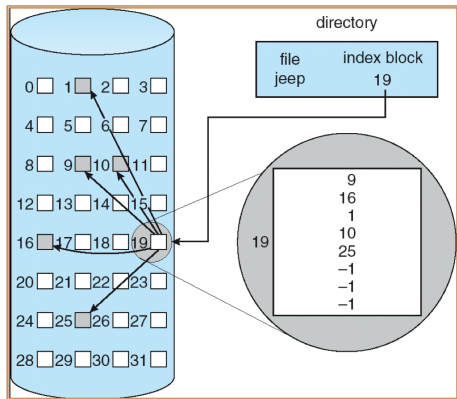
Asignación indexada

- Usamos un **bloque de índice** por fichero
- El FCB/directorio apunta al bloque de índice
- Ventajas...
- Problemas...



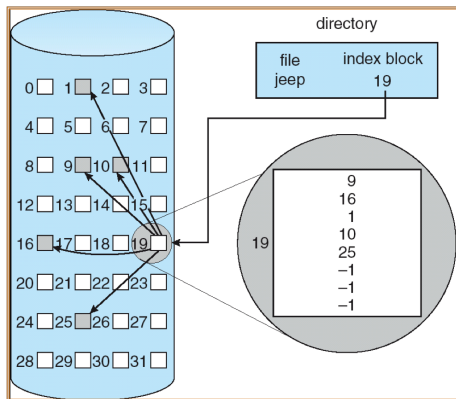
Asignación indexada

- Usamos un **bloque de índice** por fichero
- El FCB/directorio apunta al bloque de índice
- Ventajas... acceso directo óptimo
- Problemas...



Asignación indexada

- Usamos un **bloque de índice** por fichero
- El FCB/directorio apunta al bloque de índice
- Ventajas... acceso directo óptimo
- Problemas... ¿tamaño del bloque de índice?



Asignación indexada: esquemas

- Esquema enlazado
 - Cada bloque de índice ocupa un bloque de disco
 - La última entrada apunta al próximo bloque de índice (si fuera necesario más de uno)
 - Problema...
- Esquema multinivel
 - Bloque de índice de primer nivel: cada entrada apunta a un bloque de segundo nivel
 - Las entradas de los bloques de segundo nivel indican los bloques del fichero
 - Se puede generalizar a más niveles
 - Problema...

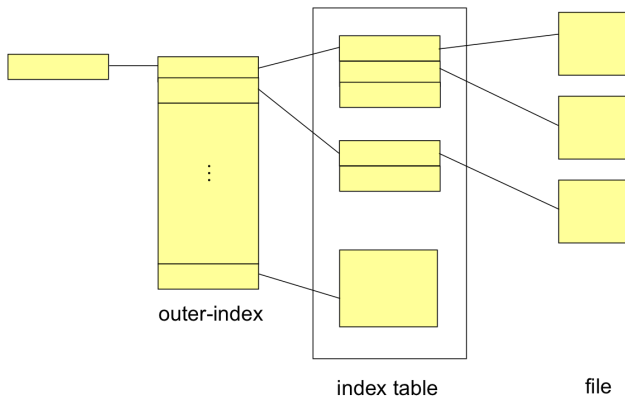
Asignación indexada: esquemas

- Esquema enlazado
 - Cada bloque de índice ocupa un bloque de disco
 - La última entrada apunta al próximo bloque de índice (si fuera necesario más de uno)
 - Problema... poco óptimo en acceso directo
- Esquema multinivel
 - Bloque de índice de primer nivel: cada entrada apunta a un bloque de segundo nivel
 - Las entradas de los bloques de segundo nivel indican los bloques del fichero
 - Se puede generalizar a más niveles
 - Problema...

Asignación indexada: esquemas

- Esquema enlazado
 - Cada bloque de índice ocupa un bloque de disco
 - La última entrada apunta al próximo bloque de índice (si fuera necesario más de uno)
 - Problema... poco óptimo en acceso directo
- Esquema multinivel
 - Bloque de índice de primer nivel: cada entrada apunta a un bloque de segundo nivel
 - Las entradas de los bloques de segundo nivel indican los bloques del fichero
 - Se puede generalizar a más niveles
 - Problema... los ficheros pequeños no necesitan dos bloques de índice, pero los usarán siempre

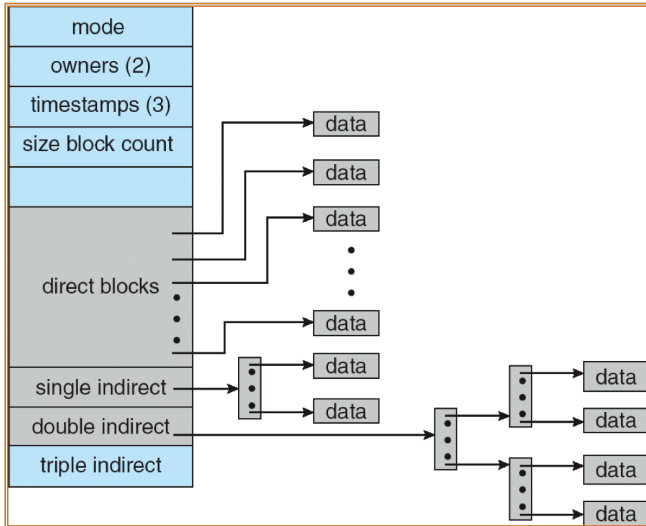
Asignación indexada: esquema multinivel



Asignación indexada: esquemas

- Esquema combinado
 - Sistema usado en Unix (UFS)
 - Las primeras $N - 3$ entradas indican los $N - 3$ primeros bloques del fichero
 - La entrada $N - 3$ apunta a un bloque de índice indirecto de un nivel
 - La entrada $N - 2$ apunta a un bloque de índice indirecto de dos niveles
 - La entrada $N - 1$ apunta a un bloque de índice indirecto de tres niveles

Asignación indexada: esquema combinado



Gestión de espacio libre

- Identificamos los bloques libres para su (re)asignación
- **Vector de bits**
 - Un mapa de bits indica si el bloque está libre (1) o no (0)
 - Problema...
- **Lista enlazada**
 - Los bloques libres contienen un puntero al próximo bloque libre
 - El sistema recuerda la cabeza de la lista
 - Problema...

Gestión de espacio libre

- Identificamos los bloques libres para su (re)asignación
- **Vector de bits**
 - Un mapa de bits indica si el bloque está libre (1) o no (0)
 - Problema...
 - ¿lo usamos directamente desde disco (ineficiente)?
 - ¿lo guardamos en memoria (es muy grande)?
- **Lista enlazada**
 - Los bloques libres contienen un puntero al próximo bloque libre
 - El sistema recuerda la cabeza de la lista
 - Problema...

Gestión de espacio libre

- Identificamos los bloques libres para su (re)asignación
- **Vector de bits**
 - Un mapa de bits indica si el bloque está libre (1) o no (0)
 - Problema...
 - ¿lo usamos directamente desde disco (ineficiente)?
 - ¿lo guardamos en memoria (es muy grande)?
- **Lista enlazada**
 - Los bloques libres contienen un puntero al próximo bloque libre
 - El sistema recuerda la cabeza de la lista
 - Problema...
 - Es ineficiente buscar bloques libres
 - Si no los buscamos, asignamos el primero (FAT):
fragmentación

Gestión de espacio libre

- **Agrupamiento**

- Optimización de la lista enlazada
- Cada bloque direcciona $N - 1$ bloques libres
- La última entrada indica el siguiente bloque libre que direccionará otros bloques

- **Recuento**

- Los bloques libres suelen estar juntos: los ficheros borrados liberan conjuntos de bloques
- Optimización: contamos con grupos de bloques a la hora de gestionarlos
- Tuplas “num. bloque - contador”