

SISTEMA DE ARCHIVOS

UNIDAD N° 2





Sistema de Archivos



1. El concepto de archivo.
2. Estructura de directorios.
3. Implementación de directorios.
4. Protección.
5. Estructura del sistema de archivos.
6. Métodos de asignación.
7. Administración del espacio libre.
8. Recuperación.



Archivo. Concepto

Definición Lógica

- Un archivo es una unidad lógica de almacenamiento, una colección de información relacionada grabada en memoria secundaria.
- **Atributos:** Nombre, Identificador, Tipo, Ubicación, Tamaño, Protección.
- **Operaciones:** Crear, Escribir, Leer, Reposicionar, Borrar, Truncar.

Estructura de Datos

- Para el SO, un archivo es una secuencia de bytes. La interpretación del contenido depende de la aplicación que lo maneja.



Archivo

Elemento central en la mayoría de las aplicaciones

Persistentes

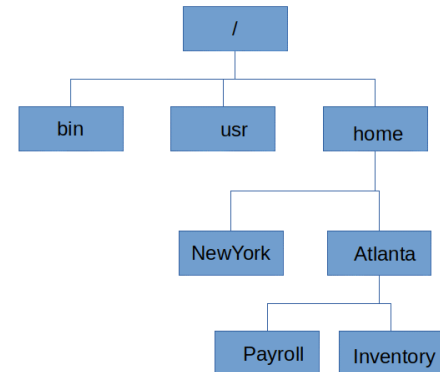
Colección de datos con nombre

Debe poder ser manipulado como una UNIDAD

Sus elementos deben poder ser manipulados individualmente

Estructura de Directorio

- El directorio es un "símbolo" que traduce nombres de archivos en sus Entradas de Control (FCB/i-nodes).
- **Tipos de Organización:**
 - **Nivel Único:** Un solo directorio para todos los usuarios.
 - **Jerárquico (Árbol):** Estructura estándar moderna (/ , C:\).
 - **Grafo Acíclico:** Permite compartir archivos mediante enlaces (links).



Directorios

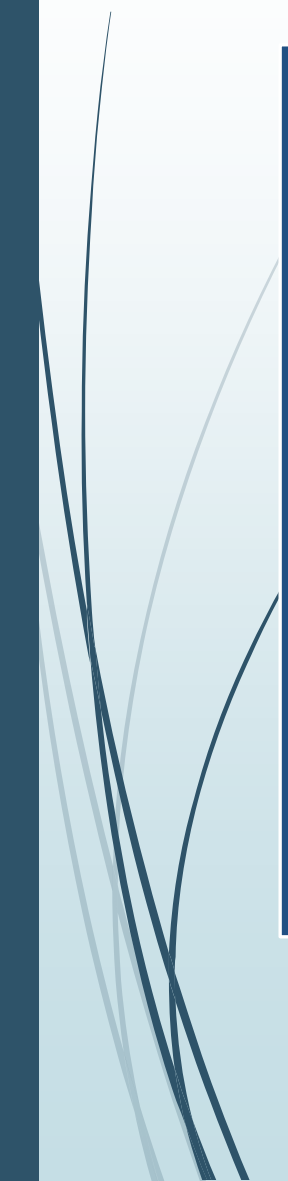
Juegos	Atributos
Correo	Atributos
Noticias	Atributos
Trabajo	atributos

*Atributos en la
entrada del directorio*

Atributos en otro lugar

Juegos		→	
Correo		→	
Noticias		→	
Trabajo		→	

→ Estructura de
datos que
contiene los
atributos



Elementos de información de un directorio

Información Básica

Nombre

Tipo

Organización

Información de direccionamiento

Volumen

Dirección inicial

Tamaño utilizado

Tamaño asignado

Información de control de acceso

Propietario

Información de acceso

Acciones permitidas

Información de uso

Fecha de creación

Creador

Fecha de último acceso

Ultimo lector

Fecha de última modificación



DIRECTORIOS



- Al abrir un archivo el S. O.:
 - Busca en su directorio el nombre del archivo.
 - Extrae los atributos y direcciones en disco.
 - Graba esta información en una **tabla de memoria real**.
 - Todas las referencias subsecuentes al archivo utilizarán la información de la memoria principal.



DIRECTORIOS. Rutas de Acceso

Ruta de Acceso Absoluta

- Consta de la ruta de acceso desde el directorio raíz hasta el archivo.
- Los componentes de la ruta de acceso se separan mediante algún carácter llamado “separador”.

Ruta de Acceso Relativa

- Se utiliza junto con el concepto de directorio de trabajo o directorio activo.
- Todos los nombres que no comiencen en el directorio raíz se toman en relación con el directorio de trabajo.



PROTECCIÓN

Control de Acceso

Mecanismo para controlar quién puede leer, escribir o ejecutar (RWX).

Grupos

Diferencia permisos entre **Dueño**, **Grupo** y **Otros**.

ACL

Listas de Control de Acceso: permiten granularidad fina por usuario específico (NTFS).



SISTEMA DE ARCHIVOS

- Parte del S. O. responsable de permitir “**compartir controladamente**” la información de los archivos.
- 



Sistema de Gestión de Archivos

Objetivos

Gestión de datos y requisitos de usuario

Validez de los datos

Optimización en el rendimiento

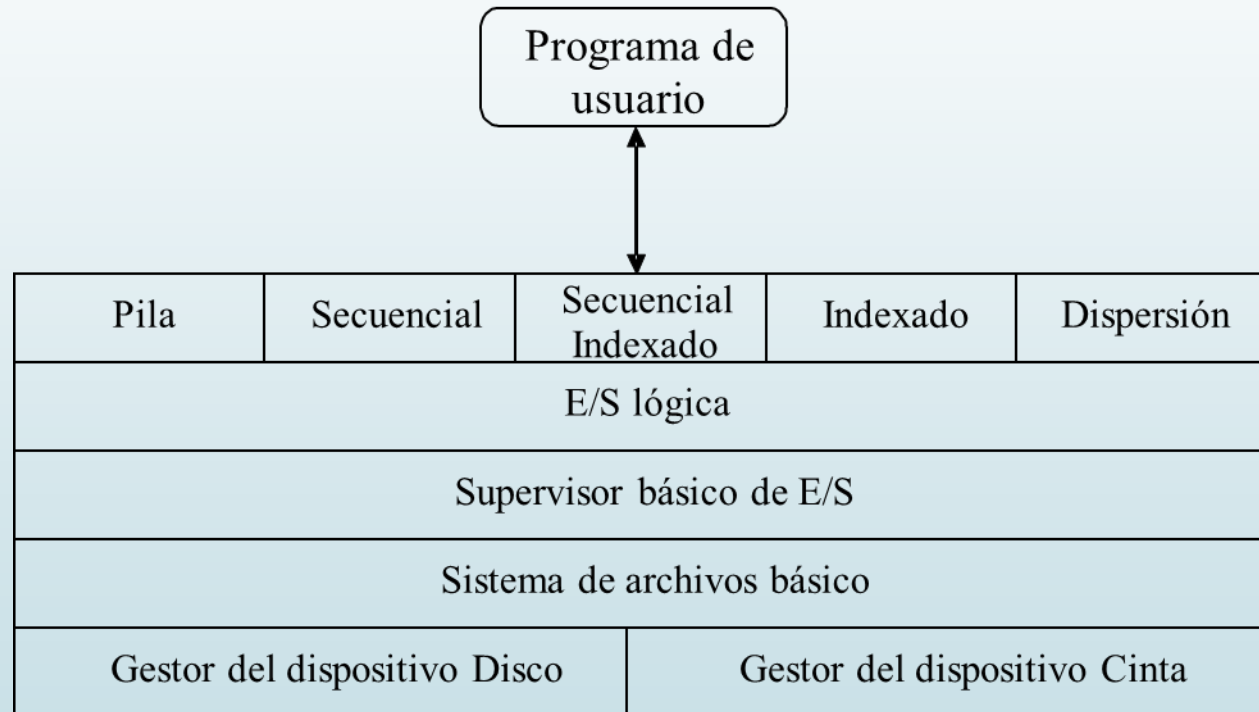
Soporte de E/S

Minimizar o eliminar la pérdida de datos

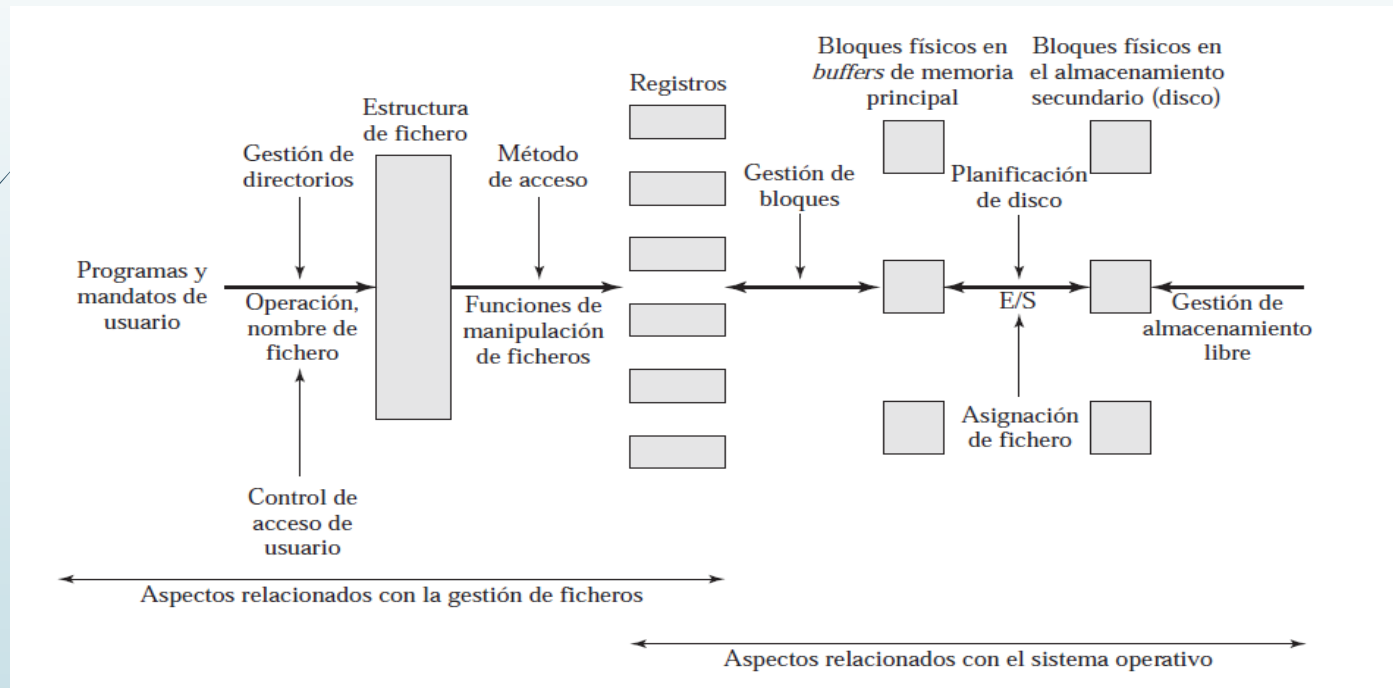
Interfaces de E/S a los procesos

Soporte de E/S en sistemas multiusuarios

Arquitectura de los Sistemas de Archivos



Funciones de la Gestión de Archivos





Organización y Acceso a los Archivos

Criterios

- Tiempo de acceso corto
- Facilidad de actualización
- Economía de almacenamiento
- Mantenimiento sencillo
- Fiabilidad

Tipos de organizaciones

- La pila
- El archivo secuencial
- El archivo secuencial indexado
- El archivo indexado
- El archivo de acceso directo o hash

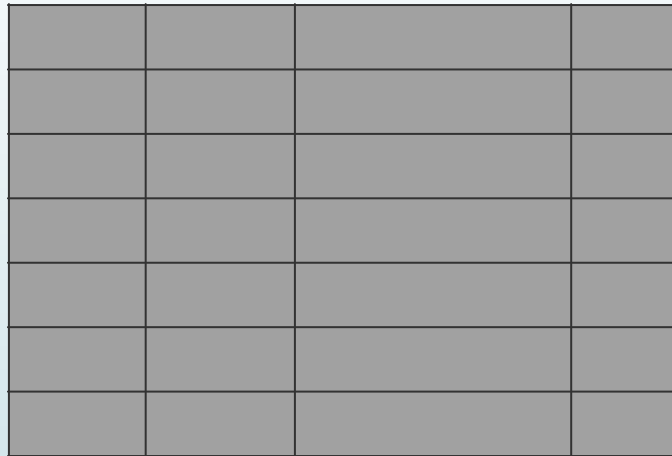
Archivos. Organización

Archivo apilado o pila



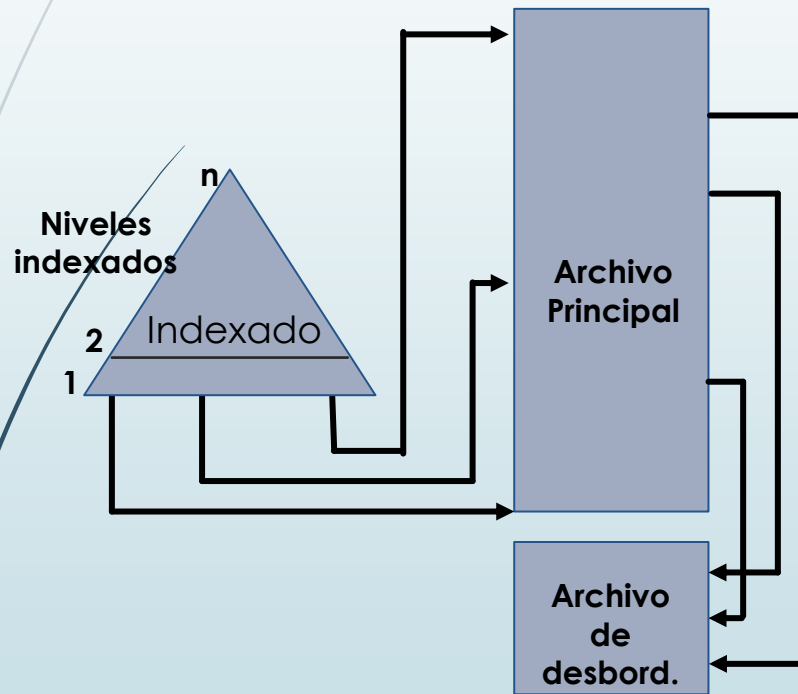
- Registro de longitud variable
- Conjunto variable de campos
- Orden cronológico

Archivo secuencial

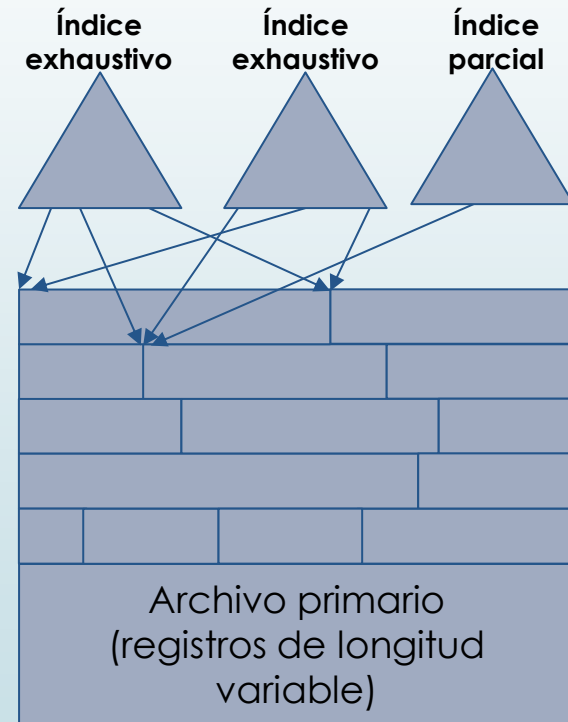


- Registro de longitud fija
- Conjunto fijo de campos en orden fijo
- Orden secuencial basado en el campo clave

Archivos. Organización

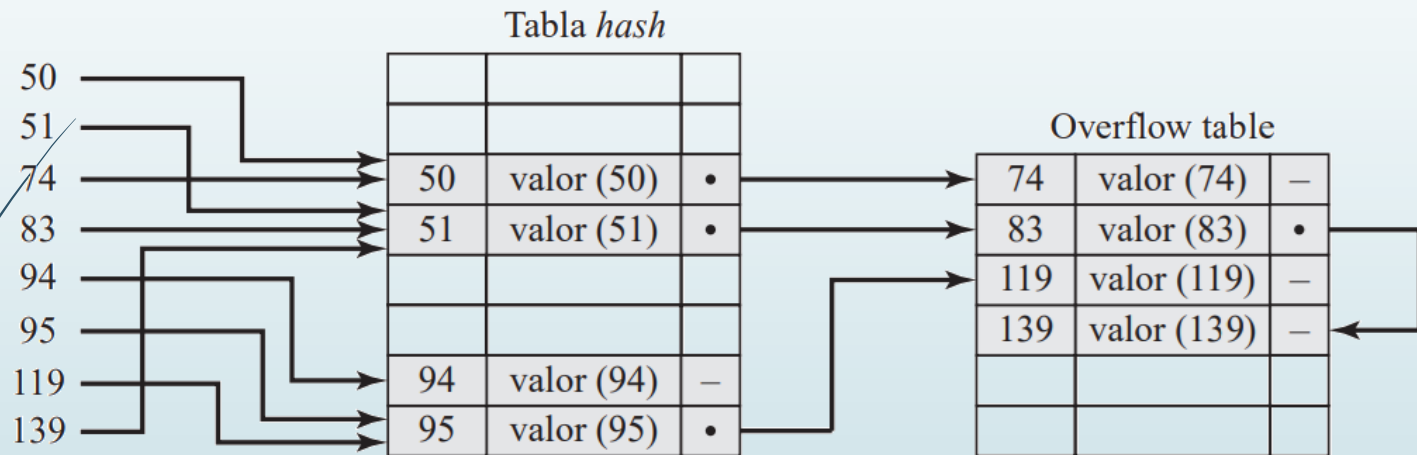


Archivo indexado secuencial



Archivo indexado

Archivos. Organización



Archivo organización directa

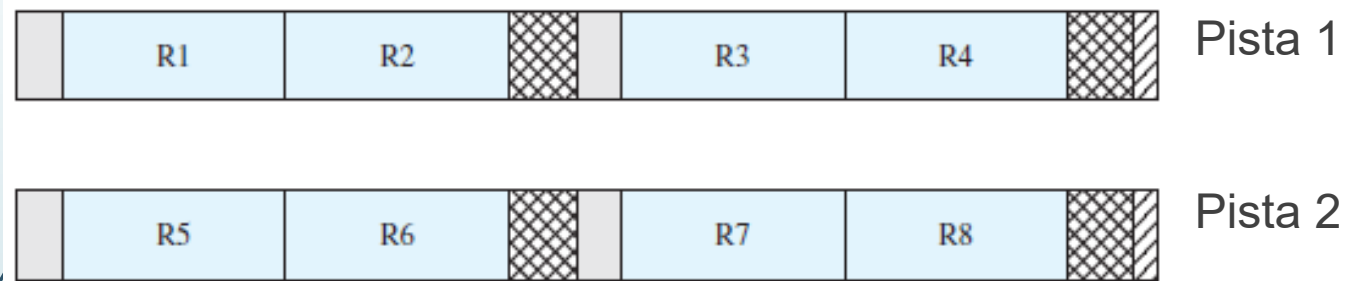


Bloques y Registros

- Tamaño de bloque vs tamaño de registro
 - Mayor tamaño del bloque, mas registros se transferirán en una operación de E/S.
 - Reducción en el número de operaciones de E/S si se accede un archivo secuencialmente
 - Transferencias innecesarias si se accede un archivo en forma aleatoria.
 - Bloques más grandes → buffers de E/S mayores.
- **Métodos**
 - Bloques fijos
 - Bloques expandidos de longitud variable
 - Bloques no expandidos de longitud variable

Bloques y Registros

Bloques Fijos

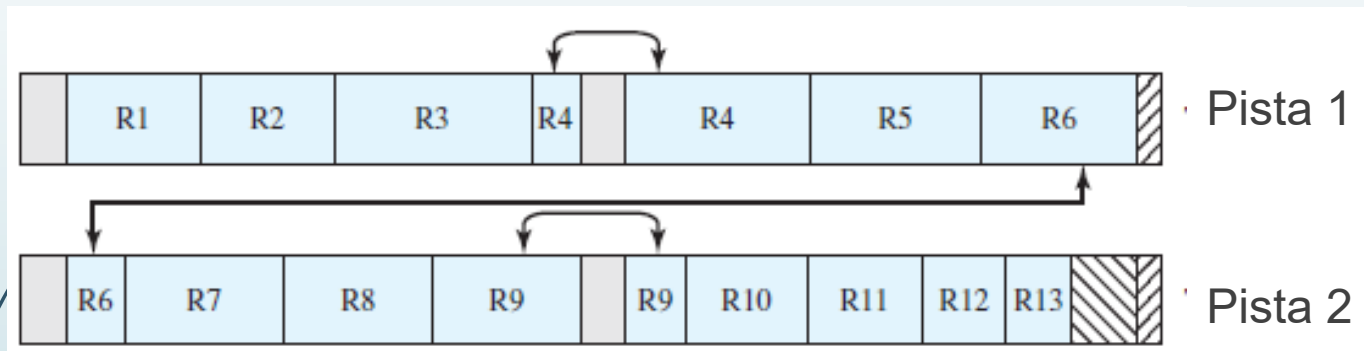


-  Datos
-  Huecos debidos al diseño hardware
-  Malgasto debido al ajuste de bloque al tamaño

-  Malgasto debido al ajuste de registro al tamaño de bloque
-  Malgasto debido a restricciones de tamaño de bloque por el tamaño de registro fijo

Bloques y Registros

Bloques Variables: extendido

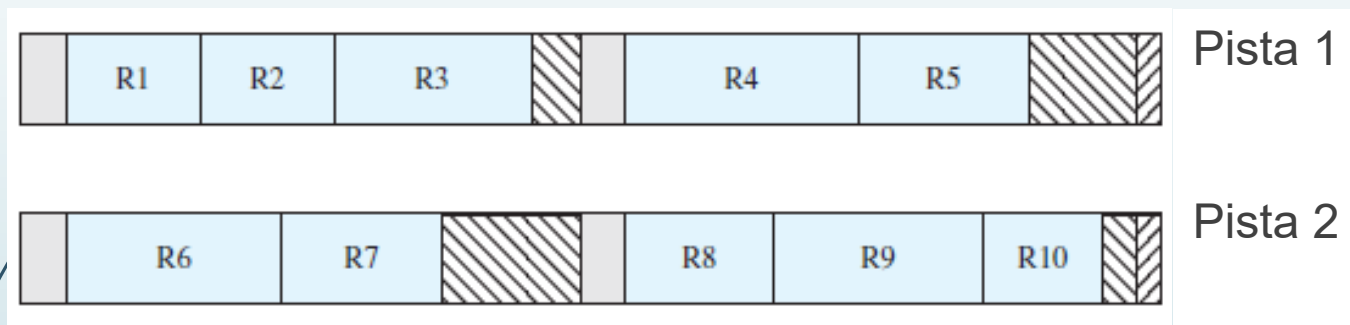


-  Datos
-  Huecos debidos al diseño hardware
-  Malgasto debido al ajuste de bloque al tamaño

-  Malgasto debido al ajuste de registro al tamaño de bloque
-  Malgasto debido a restricciones de tamaño de bloque por el tamaño de registro fijo

Bloques y Registros

Bloques Variables: no extendido



-  Datos
-  Huecos debidos al diseño hardware
-  Malgasto debido al ajuste de bloque al tamaño

-  Malgasto debido al ajuste de registro al tamaño de bloque
-  Malgasto debido a restricciones de tamaño de bloque por el tamaño de registro fijo



Gestión de Almacenamiento Secundario

- **Aspecto clave:** registro de los bloques asociados a cada archivo.
- **Aspectos:**
 - Asignación espacio de almacenamiento secundario a los archivos
 - Gestión del espacio disponible para su asignación
- **Asignación de archivos**
 - ¿Se asigna de una vez el espacio máximo requerido para el archivo?
 - ¿Cuál es el tamaño adecuado de la porción asignada?
 - ¿Qué clase de ED o tabla se utiliza para guardar traza de las porciones asignadas para un archivo?



Gestión de Almacenamiento Secundario

► Preasignación vs Asignación Dinámica

- Preasignación. Tamaño máximo del archivo declarado

► Tamaño de porción

- La contigüidad incrementa el rendimiento
- Porciones pequeñas, tablas grandes de gestión
- Porciones fijas, simplifica la reasignación de espacio
- Porciones variables, minimiza el espacio malgastado debido a la sobreasignación
- Alternativas
 - Porciones variables, grandes y contiguas.
 - Bloques



Gestión de Almacenamiento Secundario

- Estrategias de asignación
 - Primer ajuste
 - Siguiendo ajuste
 - Ajuste próximo

- Métodos de asignación de archivos
 - Contiguos
 - Encadenado
 - Indexado

Gestión de Almacenamiento Secundario



Si se tiene una unidad de asignación grande, como un **cilindro**, cada archivo, inclusive uno pequeño, ocupará todo un cilindro; con esto se desperdicia espacio de almacenamiento en disco.



Si se utiliza una unidad de asignación pequeña, como un **sector**, cada archivo constará de muchos bloques; con esto su lectura generará muchas operaciones de e/s afectando la performance.

Gestión de Almacenamiento Secundario

	Contiguos	Encadenado	Indexado	
¿Preasignación?	Necesaria	Posible	Posible	
¿Porciones de tamaño fijo o variable?	Variable	Bloques Fijos	Bloques fijos	Variable
Tamaño de porción	Grande	Pequeño	Pequeño	Medio
Frecuencia de asignación	Una vez	Pequeña a alta	Alta	Baja
Tiempo a asignar	Medio	Largo	Corto	Medio
Tamaño de tabla de asignación de archivos	Una entrada	Una entrada	Grande	Medio

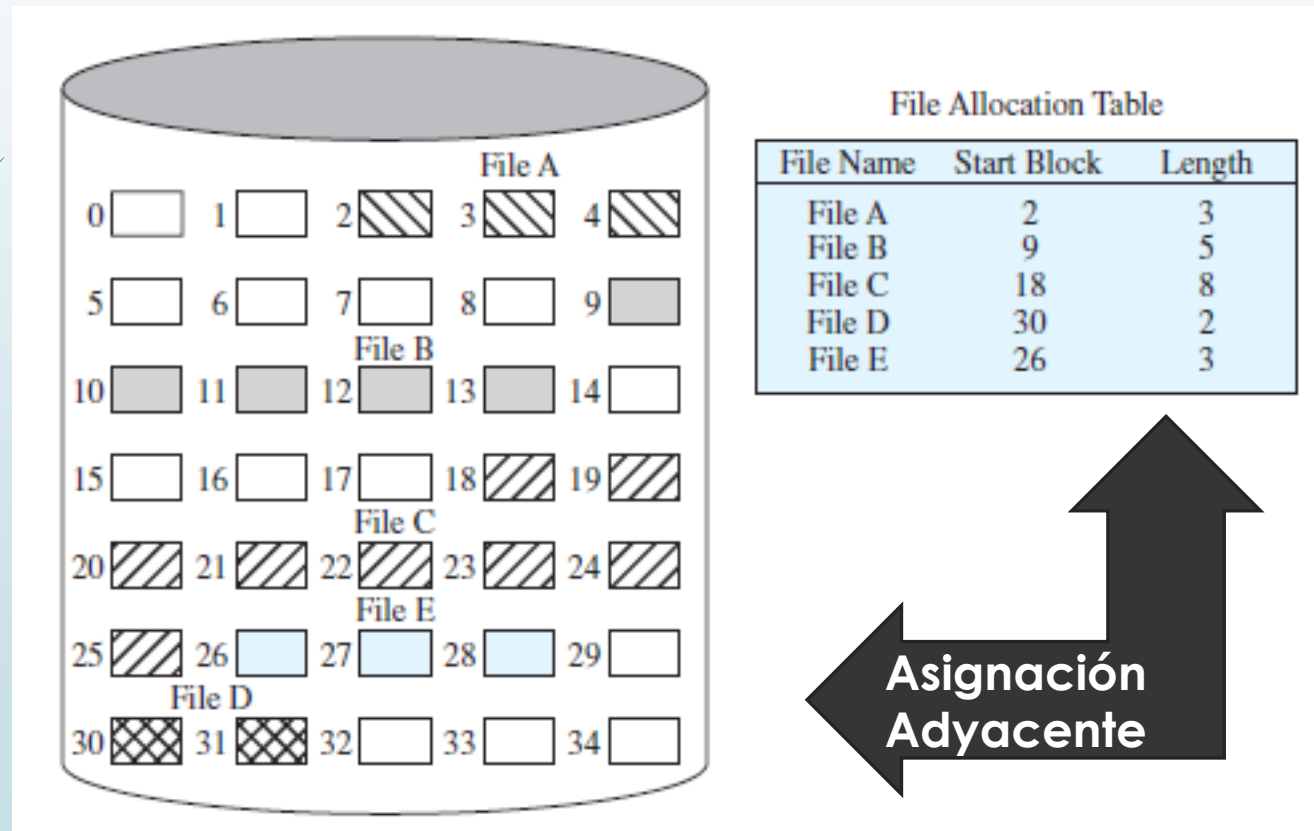


Gestión de Almacenamiento

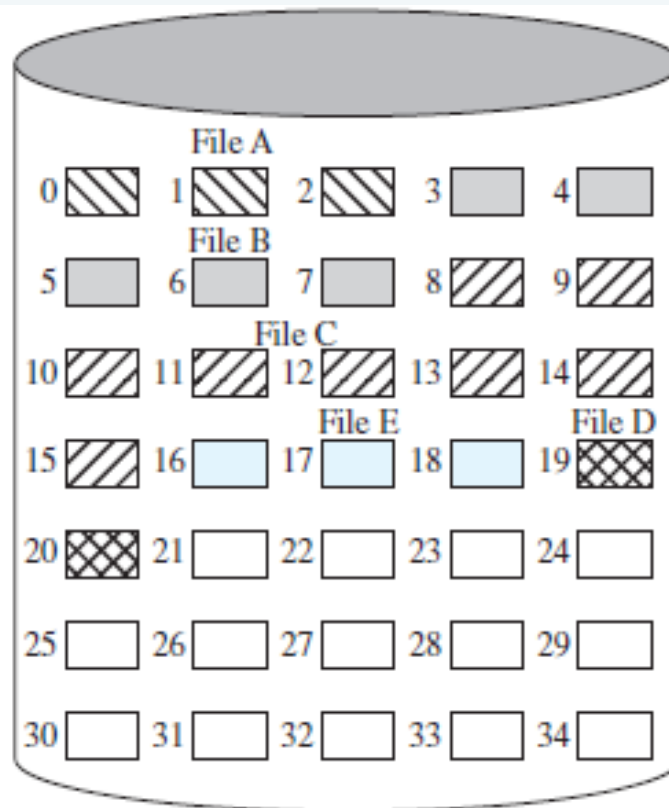
Secundario: ***Asignación adyacente***

- Los archivos son asignados a **áreas contiguas** de almacenamiento secundario.
- Las principales ventajas son:
 - **Facilidad de implantación**, ya que solo se precisa el número del bloque de inicio para localizar un archivo.
 - **Rendimiento excelente** respecto de la e / s.
- Los principales defectos son:
 - Se debe **conocer** el tamaño máximo del archivo al crearlo.
 - Produce una gran **fragmentación** de los discos.

Gestión de Almacenamiento Secundario: **Asignación Adyacente**



Gestión de Almacenamiento Secundario: *Asignación Adyacente*



File Allocation Table

File Name	Start Block	Length
File A	0	3
File B	3	5
File C	8	8
File D	19	2
File E	16	3

Asignación
Adyacente (post-
compactación)



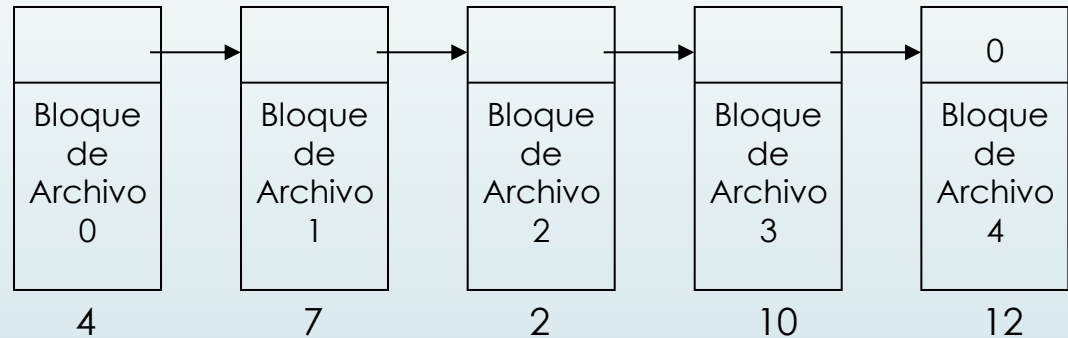
Gestión de Almacenamiento Secundario: *Asignación Ligada*

- Cada archivo se mantiene como una **lista ligada de bloques en disco**
- La **primera palabra** de cada bloque se utiliza como **apuntador** a la siguiente. El resto del bloque contiene datos.
- Las principales ventajas son:
 - No se pierde espacio por fragmentación.
 - Es suficiente que la entrada del directorio guarde sólo la dirección en disco del primer bloque. El resto se puede encontrar a partir de esto.
- Las principales desventajas son:
 - El acceso aleatorio es muy lento.
 - La cantidad de espacio para almacenamiento de datos en un bloque ya no es una potencia de 2, debido al apuntador que ocupa unos cuantos bytes.

Gestión de Almacenamiento Secundario: *Asignación Ligada*

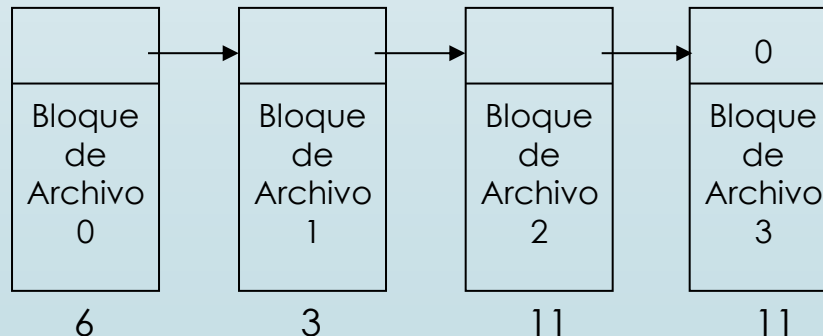
Bloque Físico:

Archivo A:

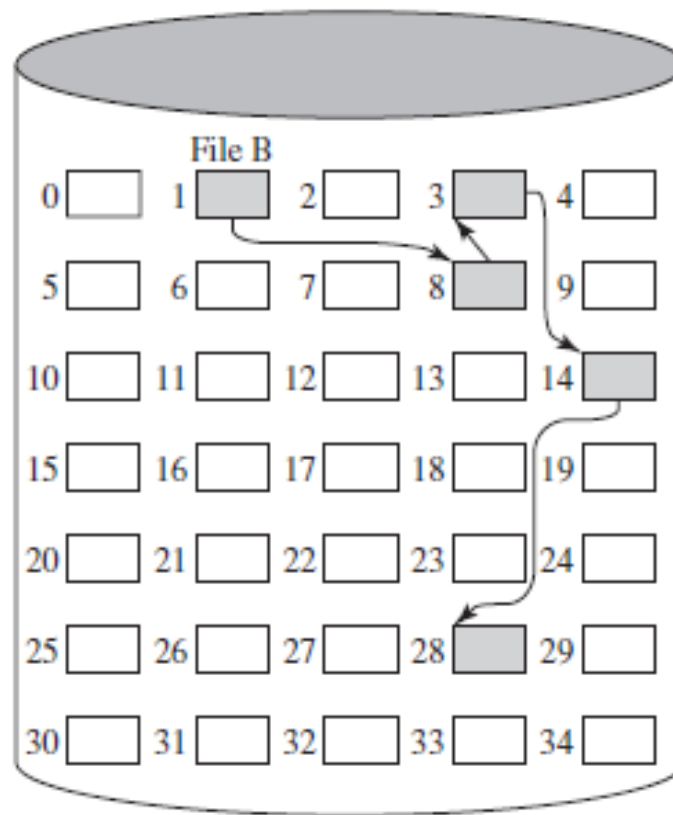


Archivo B:

Bloque Físico:



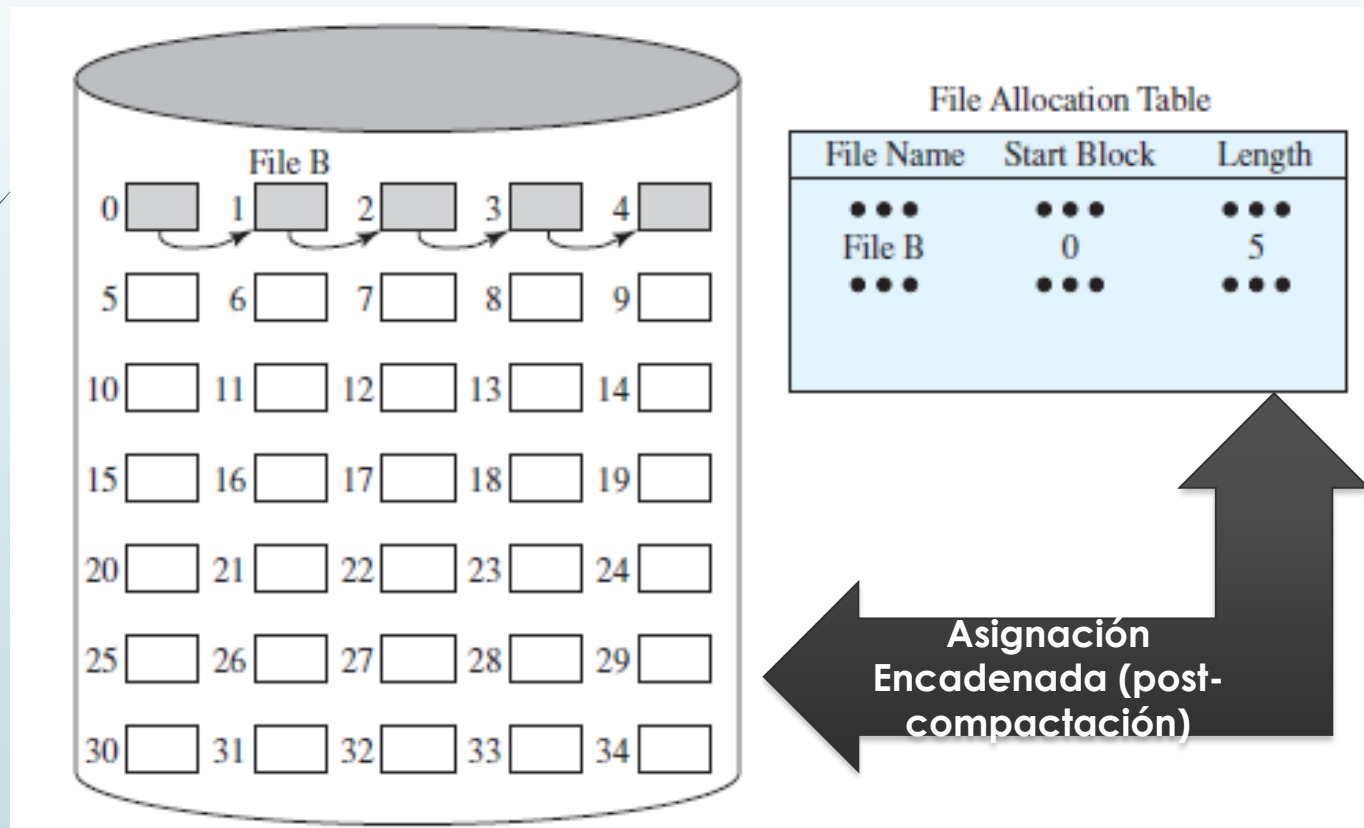
Gestión de Almacenamiento Secundario



File Allocation Table		
File Name	Start Block	Length
...	...	...
File B	1	5
...	...	...



Gestión de Almacenamiento Secundario



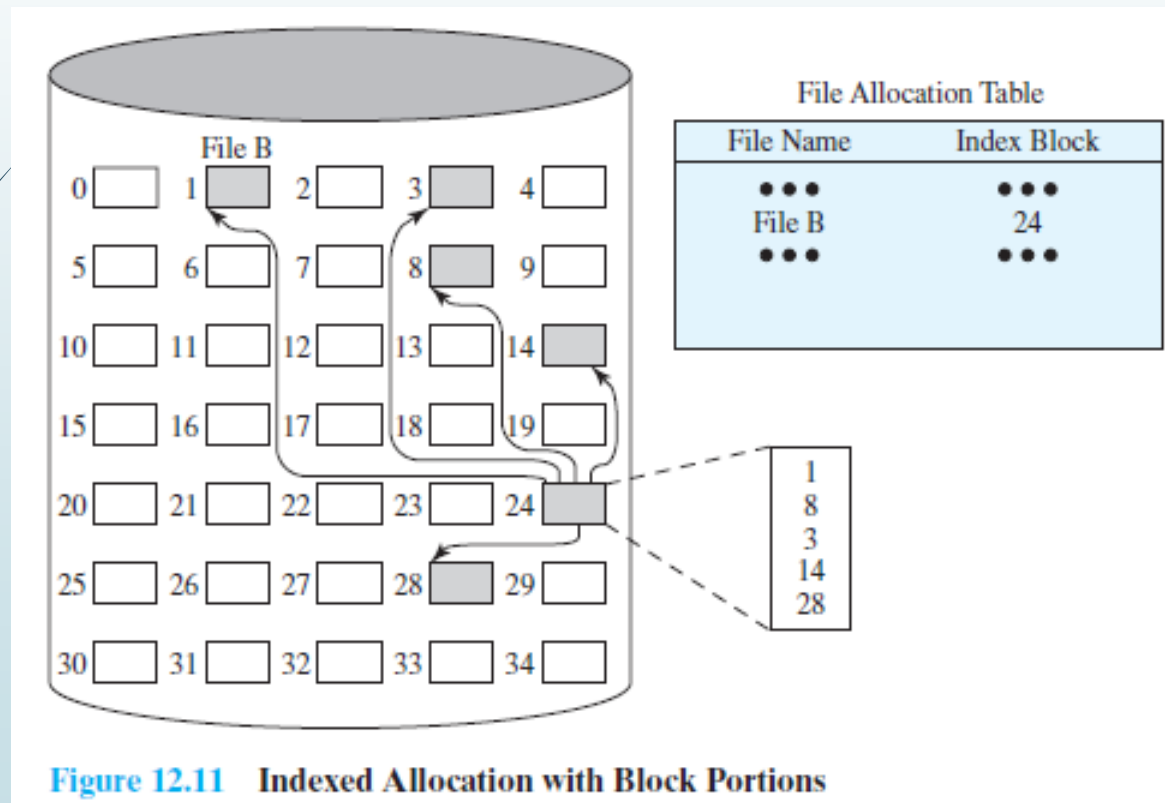


Gestión de Almacenamiento

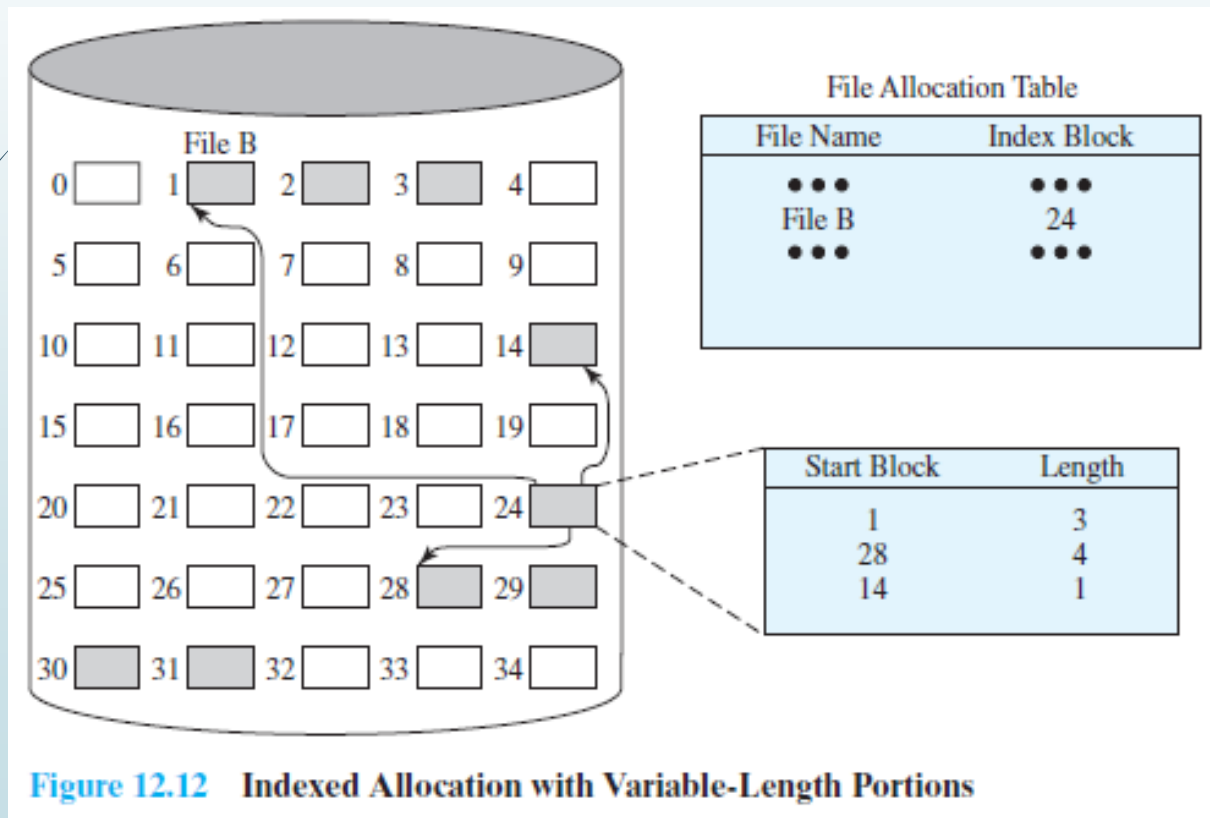
Secundario: ***Asignación Ligada e Indizada***

- Las desventajas del método anterior se pueden eliminar si se toma la palabra del apuntador de cada bloque del disco y se lo coloca en una tabla o índice de la memoria.
- Las principales ventajas son:
 - Todo el bloque está disponible para los datos.
 - El acceso aleatorio es mucho mas sencillo.
- La principal desventaja es:
 - Toda la tabla debe estar en memoria todo el tiempo para que funcione.

Gestión de Almacenamiento Secundario



Gestión de Almacenamiento Secundario





Gestión de Almacenamiento

Secundario: **Nodos-i**

- Asocia a cada archivo una pequeña tabla, llamada nodo-i (nodo índice), la cual contiene los atributos y direcciones en disco de los bloques del archivo.
- Las 1ras direcciones en disco se almacenan en el propio nodo-i, de manera que para los archivos pequeños, toda la información necesaria está contenida en el nodo-i, el cual se traslada del disco a la memoria principal al abrir el archivo.
- Para archivos grandes, el nodo-i contiene la dirección de un bloque en el disco llamado bloque simplemente indirecto. Éste puede contener otra dirección en el nodo-i, el bloque doblemente indirecto, el cual contiene la dirección de un bloque que presenta una lista de los bloques simplemente indirectos.
- Cada uno de estos bloques, apunta a unos centenares de bloques de datos, pudiéndose utilizar un bloque triplemente indirecto.

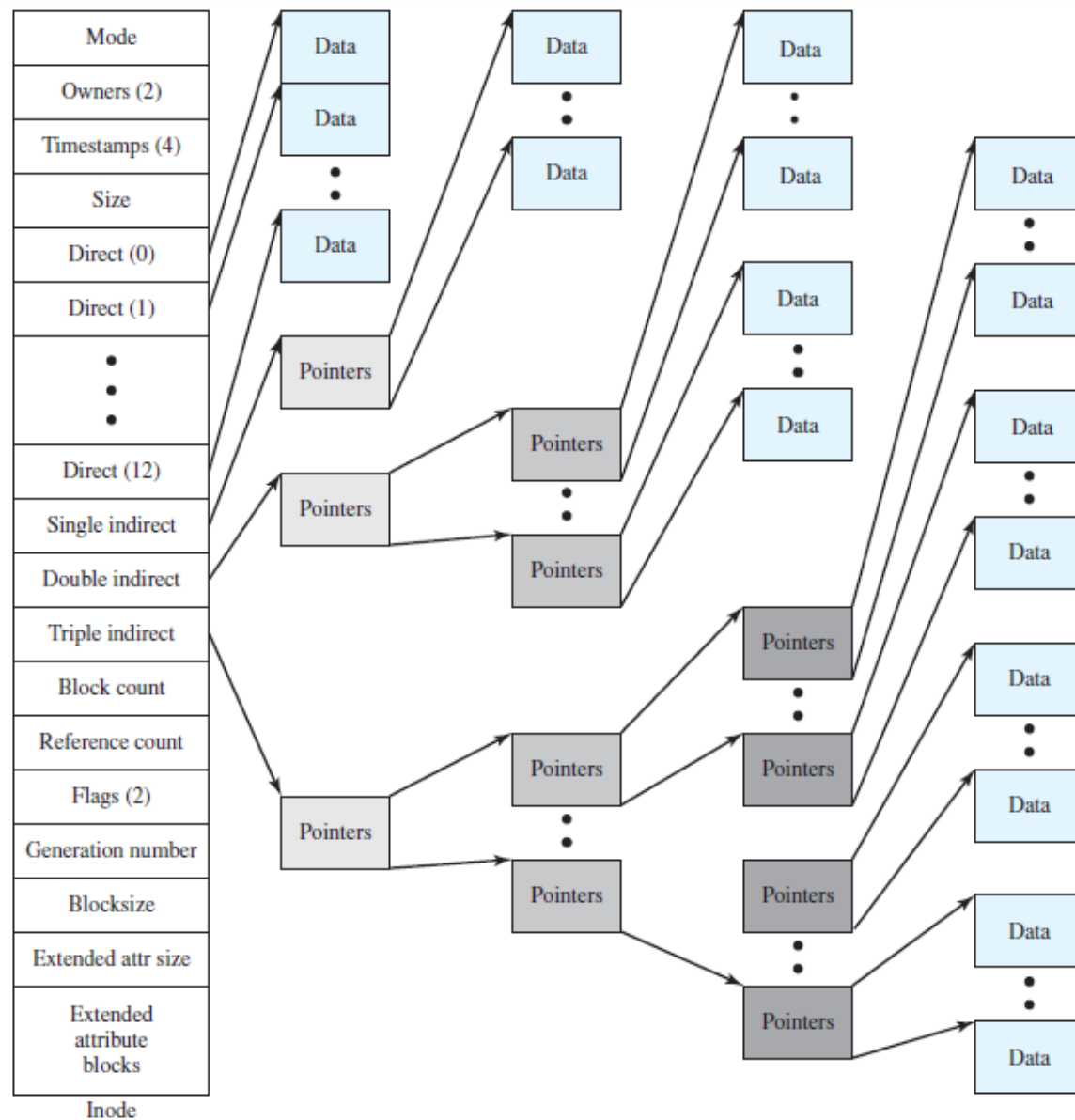


Figure 12.14 Structure of FreeBSD inode and File

Gestión de Almacenamiento Secundario

Capacidad de un archivo UNIX

Nivel	Número de bloques	Número de bytes
Directo	10	10K
Simple Indirecto	256	256K
Doble Indirecto	$256 \times 256 = 65K$	65M
Triple Indirecto	$256 \times 65K = 16M$	16G

Gestión de Almacenamiento Secundario

Gestión de
Espacio Libre
(métodos)

Un **mapa de bits** o
tabla de bits

La **lista de
bloques libres**
como lista ligada.

Gestión de Almacenamiento Secundario. **Mapa de Bits**

- Un disco con “ n ” bloques necesita un mapa de bits con “ n ” bits.
- Los bloques libres se representan con “1” y los asignados con “0” (o viceversa).
- Generalmente este método es preferible cuando existe espacio suficiente en la memoria principal para contener *completo el mapa de bits*.
- La cantidad de memoria (en bytes) requerida para un mapa de bits de bloques es:

$$\frac{\text{tamaño de disco en bytes}}{8 * \text{tamaño de bloques del sistema de archivos}}$$

Gestión de Almacenamiento

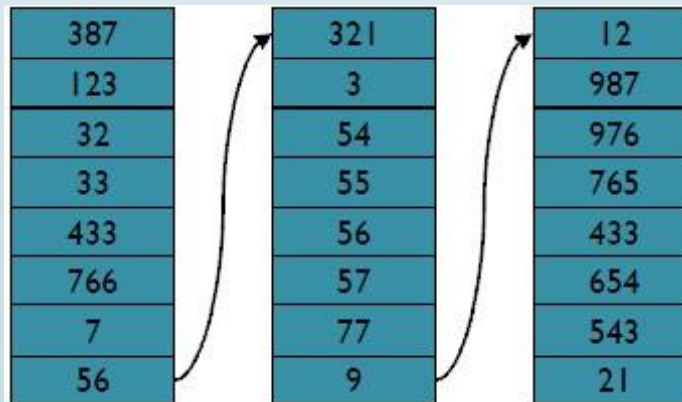
Secundario: *Mapa de bits*

0	1	0	0	0	0	0	0	1	1	0	1	0	1	0	1
0	1	0	0	0	1	1	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	1	0	0	1	1	0	1	0	1	0
0	0	0	0	1	0	0	0	1	1	0	1	0	0	1	1
0	1	0	0	0	1	0	0	1	1	0	1	0	0	0	0
0	1	1	0	1	0	0	1	0	0	0	1	0	0	0	1

Gestión de Almacenamiento Secundario

► Lista ligada de bloques de disco:

- Cada bloque contiene tantos números de bloques libres como pueda.
- Los bloques libres se utilizan para contener a la lista de bloques libres.



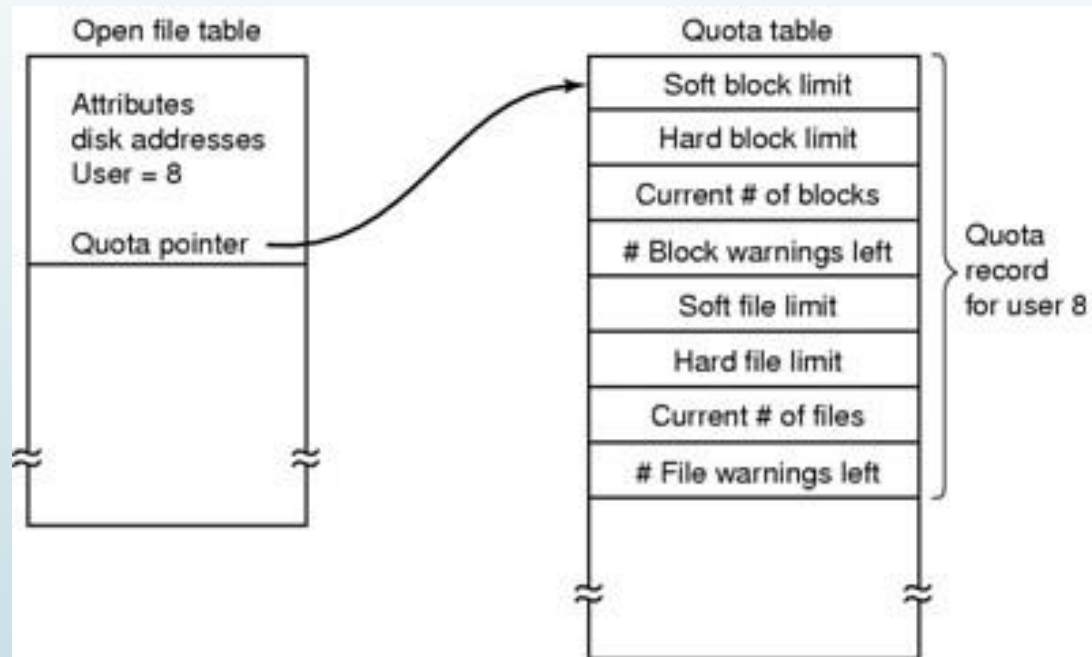


Gestión de Almacenamiento Secundario

DISK QUOTES

- Para evitar que los usuarios se apropien de un espacio excesivo en disco, los S. O. multiusuario proporcionan generalmente un mecanismo para establecer las cuotas en el disco.
- La idea es que:
 - Un administrador del sistema asigne a cada usuario una proporción máxima de archivos y bloques.
 - El S. O. garantice que los usuarios no excedan sus cuotas

Disk Quotas





Disk Quotes



- Un mecanismo utilizado es el siguiente:
 - Cuando un usuario abre un archivo:
 - Se localizan los atributos y direcciones en disco.
 - Se colocan en una tabla de archivos abiertos en la memoria principal.
 - Uno de los atributos indica el propietario del archivo; cualquier aumento del tamaño del archivo se carga a la cuota del propietario.
 - Una segunda tabla contiene el registro de las cuotas para cada uno de los usuarios que tengan un archivo abierto en ese momento, aún cuando el archivo lo haya abierto otro usuario.



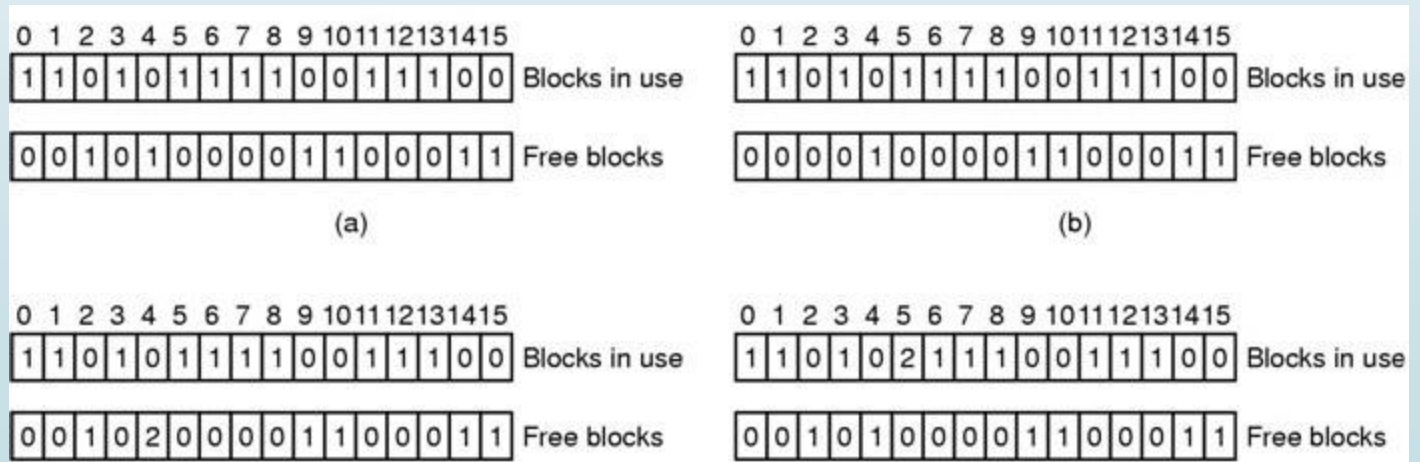
Disk Quotes

- Un mecanismo utilizado es el siguiente:
 - Cuando se escribe una **nueva entrada en la tabla de archivos abiertos**:
 - Se introduce un apuntador al registro de la cuota del propietario para localizar los límites.
 - Cuando se **añade un bloque a un archivo**:
 - Se incrementa el total de bloques cargados al propietario.
 - Se verifica este valor contra los límites estricto y flexible (el primero no se puede superar, el segundo sí).
 - También se verifica el número de archivos

Estados del Sistema de Archivos

- a) Consistente
- b) Bloque perdido

- c) Bloque duplicado en la lista libre
- d) Bloque de datos duplicado





Confiabilidad Del Sistema De Archivos

- Es necesario **proteger** la información alojada en el sistema de archivos, efectuando los resguardos correspondientes.
- De esta manera se evitan las consecuencias generalmente catastróficas de la **pérdida** de los sistemas de archivos.
- Las pérdidas se pueden deber a problemas de hardware, software, hechos externos, etc.

Confiabilidad Del Sistema De Archivos

Manejo de un bloque defectuoso:

- Se utilizan soluciones por hardware y por software.
- La **solución en hardware:**
 - Consiste en dedicar un sector del disco a la *lista de bloques defectuosos*.
 - Al inicializar el controlador por primera vez:
 - Lee la “*lista de bloques defectuosos*”.
 - Elige un bloque (o pista) de reserva para reemplazar los defectuosos.
 - Registra la asociación en la lista de bloques defectuosos.
 - En lo sucesivo, las solicitudes del bloque defectuoso utilizarán el de repuesto



Confiabilidad Del Sistema De Archivos

► La **solución en software**:

- Requiere que el usuario o el sistema de archivos construyan un *archivo con todos los bloques defectuosos*.
- Se los elimina de la *“lista de bloques libres”*.
- Se crea un *“archivo de bloques defectuosos”*:
 - Esta constituido por los bloques defectuosos.
 - No debe ser leído ni escrito.
 - No se debe intentar obtener copias de respaldo de este archivo

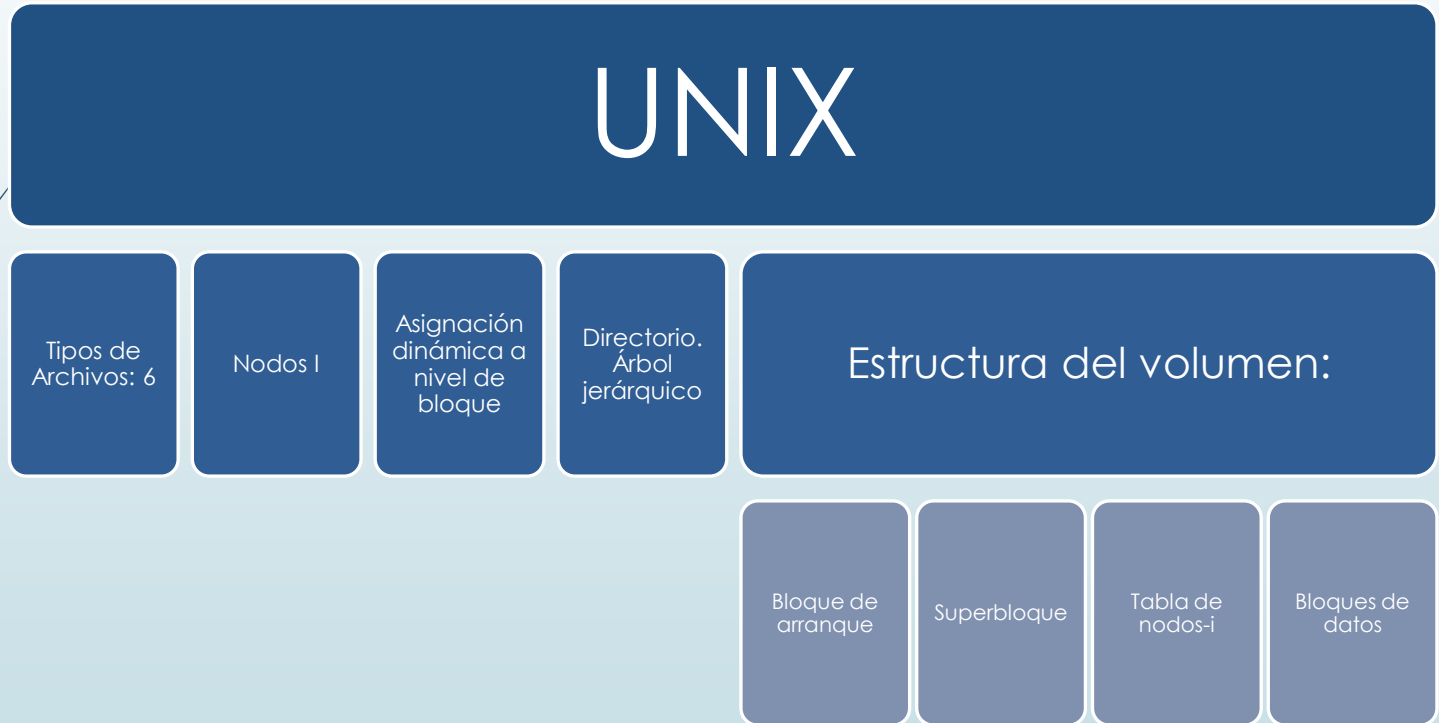


Confiabilidad Del Sistema De Archivos

Respaldos (copias de seguridad o de back-up):

- Es muy importante respaldar los archivos con frecuencia.
- Los respaldos pueden consistir en efectuar copias completas del contenido de los discos (flexibles o rígidos).
- Una estrategia de respaldo consiste en dividir los discos en *áreas de datos* y *áreas de respaldo*, utilizándolas de a pares
- Otra estrategia es el vaciado por incrementos o *respaldo incremental*

Casos de Estudio



Casos de Estudio

WINDOWS

Tipos de Archivos:
12

NTFS

Asignación
dinámica
a nivel de
agrupacio
nes (Min:
512 bytes)

Directorio.
Árbol
jerárquico

Estructura del volumen:

Sector de
arranque
de
partición

Tabla
maestra
de
ficheros

Ficheros
de
sistema

Área de
ficheros



Fin