



**Ingeniería en Desarrollo de Software**  
**3<sup>er</sup> semestre**

Programa de la Unidad didáctica:  
**Programación de sistemas operativos**

**Unidad 2. Administrador de sistemas de archivos**

Clave:

Ingeniería:                    TSU:  
**15142317**                    **16142317**

Ciudad de México, julio 2025

**Universidad Abierta y a Distancia de México**





### Índice

Unidad 2. Administrador de sistemas de archivos .....	3
Presentación de la unidad .....	3
Logros .....	4
Competencia específica .....	4
2.1. Sistemas de archivos.....	4
2.1.1. Concepto.....	6
2.1.2. Jerarquía de datos.....	8
2.2. Tipo de archivos.....	11
2.2.1. Real.....	14
2.2.2. Virtual.....	15
2.2.3. Componentes.....	16
2.3. Organización del sistema de archivos, lógica y física .....	17
2.3.1. Operaciones con archivos .....	20
2.3.2. Implementación de archivos.....	22
cierre de la unidad .....	26
Para saber más.....	26
Fuentes de consulta.....	27



## Unidad 2. Administrador de sistemas de archivos

### Presentación de la Unidad

A lo largo de esta unidad se expondrá la forma en que el sistema operativo maneja el sistema de archivos, los tipos de archivos, y cómo son organizados de manera lógica y física.

El sistema de archivos, como se verá más adelante, está encargado de la administración de la información en: acceso, verificación y manipulación; logrando con ello ser un método de almacenamiento y organización. Además de permitir encontrar y acceder más rápido a la información. Un sistema de archivos es un conjunto de tipo de datos abstractos, que son implementados para el almacenamiento, la organización jerárquica, la manipulación, el acceso, el direccionamiento y la recuperación de datos.

Otro elemento importante, que se verá en esta unidad es la diferencia entre archivo real y virtual.

Un **archivo virtual** es un archivo que se almacena en un sistema de almacenamiento en la nube, por ejemplo, Dropbox, Skydrive, y se puede visualizar cuando se consulta un archivo y genera una copia del original hasta que sea guardado físicamente. Otros tipos de archivos virtuales son aquellos que genera el sistema de manera temporal y se generan mientras se está en la ejecución de un proceso con la finalidad de almacenar información temporal y organizarla.

Por otra parte, un **archivo real** es todo aquel archivo, programa o datos que se guarda en la computadora, y que al apagar el sistema permanece, es decir, todos aquellos archivos que ocupan una parte física en bytes en la computadora.

Para finalizar la unidad se abordarán las operaciones que son permitidas por parte de los archivos y cómo se implementan.



### Logros

Al término de esta unidad lograrás:

- Asociar los sistemas de archivos.
- Comparar archivo real, virtual y sus componentes.
- Distinguir las operaciones entre archivos.
- Distinguir la implementación de los datos de archivos.

### Competencia específica

- Identificar el proceso que siguen los archivos para la administración y buen manejo de los mismos, con el análisis lógico que sigue el sistema.

### 2.1. Sistemas de archivos

Para los usuarios, el aspecto más importante de un sistema de archivos es lo que de éste se ve; es decir, qué constituye un archivo, cómo se nombran y protegen los archivos, qué operaciones pueden efectuarse con los archivos.

El sistema de software que proporciona a los usuarios y a las aplicaciones servicios relativos al empleo de archivos debe cumplir con las necesidades de gestión de datos y con las necesidades de almacenamiento del usuario; se deben cumplir los siguientes objetivos:

- Acreditar que los datos de los archivos sean válidos.
- Optimizar el rendimiento.
- Otorgar soporte E/S para la variedad de tipos de dispositivos de almacenamiento.
- Minimizar o eliminar la posibilidad de pérdida de los datos.
- Implantar un conjunto estándar de rutinas de interfaz de E/S.
- Proporcionar un soporte de E/S para múltiples usuarios en los sistemas multiusuario.

Un **sistema de archivos** se conforma de los métodos y estructuras, que son los datos que un sistema operativo utiliza para seguir la pista de los archivos de un disco o partición, y las formas en que se organizan los archivos en el disco (Niño, 2011). Si deseas conocer más sobre



los sistemas de archivos y sistema de archivos que se utilizan en cada uno de los sistemas operativos existentes en el mercado es recomendable que consultes las páginas 44 y 45 de la obra *Sistemas operativos monopuestos* del autor Niño (2011).

El concepto de sistema de archivos, también se puede utilizar para hacer referencia a una partición o disco que se utiliza para almacenamiento de la información o el tipo del sistema de archivos que utiliza.

El disco duro se encuentra particionado o dividido en sectores, en cada uno de ellos se almacena información, si el sistema de archivos decide guardar información en un sector que ya se encuentra ocupado, puede ser un grave problema, pero más aún se incrementa este problema si quien lo ocupa es parte del sistema mismo, creando un grave conflicto que ocasionaría que el sistema colapse por completo.

Duran (2000) afirma que construir un sistema de archivos consiste en que el mismo sistema de archivos organice y distribuya los datos en áreas accesibles del disco duro para poder trabajar con la información almacenada sin ningún tipo de problema. Esto es así porque el sistema operativo debe saber en dónde se encuentra cada archivo.

Los archivos están administrados por el sistema operativo. La manera en que están estructurados, nombrados, cómo se accede a ellos, cómo se utilizan, cómo se protegen e implementan son características que se utilizan para el diseño de los sistemas operativos. A esta parte del sistema operativo que gestiona archivos, comúnmente se le conoce como **sistema de archivos**.

Para un usuario, el aspecto de mayor importancia dentro de un sistema de archivos es su apariencia; en otras palabras, es lo que conforma un archivo, cómo se nombran y se protegen los archivos, las operaciones que se permiten, etc.

En resumen, los sistemas de archivos son diseñados para que el sistema operativo se encargue de leer y consultar los datos del disco. Por lo que, el usuario no necesariamente se entera de detalles de dónde y cómo se almacenará la información, o el funcionamiento de los discos, ya que como se mencionó anteriormente, lo hace en particiones del disco duro, siendo convertido ese proceso en un mecanismo de abstracción.



La forma en la que se da nombre a los archivos que se manejan podría ser la característica más importante del mecanismo de abstracción, una vez que el proceso crea el archivo, se le asigna el nombre. Cuando el proceso termina, el archivo sigue existiendo y otros programas pueden tener acceso a él, utilizando su nombre.

Varios sistemas de archivos administran los nombres de éstos en dos partes separadas por un punto; a la parte secuencial del punto se le nombra extensión del archivo y normalmente especifica el tipo del archivo, por ejemplo, en MS-DOS, los nombres de los archivos cuentan con uno o hasta ocho caracteres, más una extensión opcional de uno a tres caracteres. En Unix, se deja la extensión opcional de contar con dos o más extensiones y son especificaciones que se dejan a criterio del mismo usuario. En la siguiente tabla se presenta un listado de las extensiones más comunes de los sistemas de archivos y su descripción.

Extensión	Significado
archivo.bak	Archivo de respaldo
archivo.c	Programa fuente en C
archivo.gif	Imagen en Formato de Intercambio Gráfico de CompuServe
archivo.hlp	Archivo de ayuda
archivo.html	Documento en Lenguaje de Marcado de Hipertexto de World Wide Web
archivo.jpg	Imagen fija codificada con el estándar JPEG
archivo.mp3	Música codificada con el formato de audio MPEG capa 3
archivo.mpg	Película codificada con el estándar MPEG
archivo.o	Archivo objeto (salida del compilador, todavía sin enlazar)
archivo.pdf	Archivo en Formato de Documento Portable
archivo.ps	Archivo PostScript
archivo.tex	Entrada para el programa de formateo TEX
archivo.txt	Archivo de texto general
archivo.zip	Archivo comprimido

Tabla de extensiones de algunos sistemas de archivos

Tomada de Tanenbaum (2003, p. 381).

### 2.1.1. Concepto

Los archivos proporcionan una forma de almacenar información en el disco y de leerla después. Esto debe hacerse de tal manera que el usuario no tenga que ocuparse de los detalles de cómo y dónde se almacena la información, ni de cómo funcionan realmente los



discos. Las reglas para nombrar archivos varían de un sistema a otro, pero todos los sistemas operativos permiten cadenas de uno a ocho caracteres como nombres de archivos válidos.

Los archivos por su contenido pueden ser numéricos, alfanuméricos o binarios, pueden existir archivos de formato libre, como son los archivos de texto, o bien los archivos pueden estar formateados de forma rígida.

La información que contiene un archivo es definida por el usuario. Pueden almacenar muchos tipos de información, por ejemplo, programas puente, programas objeto, programas ejecutables, datos numéricos, texto, registros varios, imágenes, archivos de audio, etc. Para esto cada tipo de archivo tiene su propia estructura definida.

De acuerdo con Stallings (2005) al tratar el tema de archivos, básicamente, se utilizan cuatro términos comunes:

**Campo.** Es el elemento de datos básico, un campo individual contiene un valor único, como el apellido de una persona, una fecha o el valor leído por un sensor. Se caracteriza por su longitud y por el tipo de datos.

**Registro.** Es una colección de campos relacionados que pueden tratarse como una unidad en algunos programas de aplicación.

**Archivo.** Es una colección de información similar, con un nombre, la cual se guarda en un almacenamiento secundario. Los usuarios y las aplicaciones administran al archivo como una entidad única y se refieren a él por un nombre, los archivos tienen nombres únicos y pueden crearse y eliminarse”.

**Base de datos.** Es una colección de archivos relacionados, el aspecto básico de una base de datos se basa en las relaciones que existen entre los elementos de datos, son explícitas, y la base de datos está diseñada para ser utilizada por varias aplicaciones diferentes.

Un **archivo fuente** es una secuencia de subrutinas y funciones, cada una de las cuales está a su vez organizada como una serie de declaraciones, seguida de instrucciones ejecutables. Un



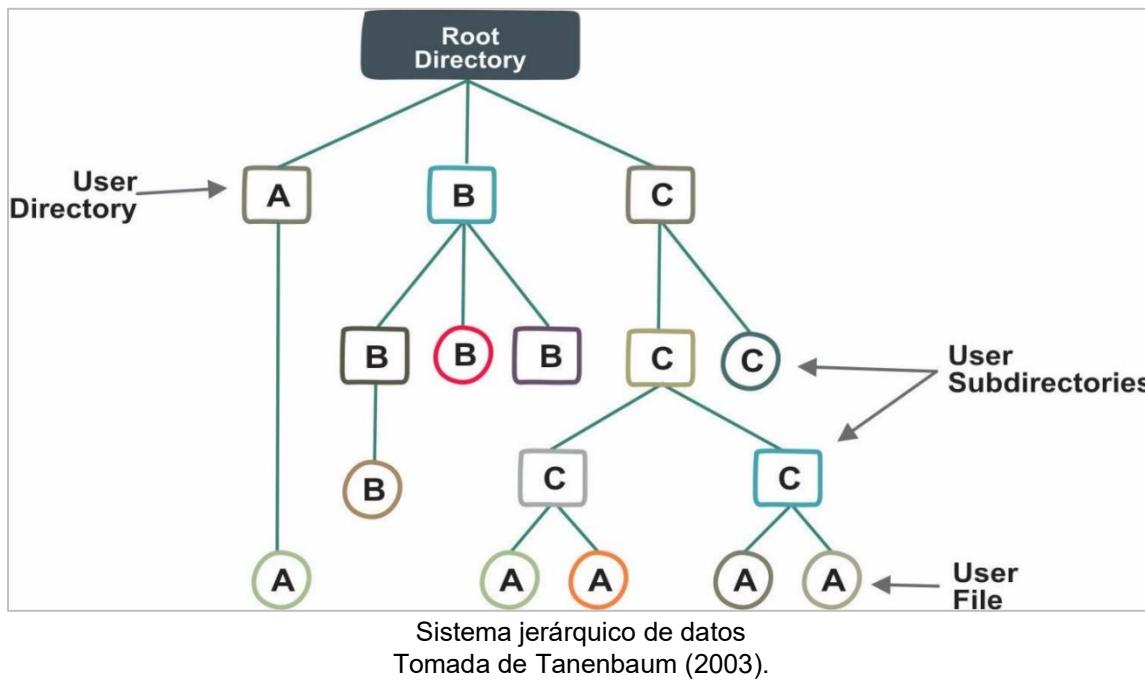
**archivo objeto** es una secuencia de bytes organizado en bloques que el programa montador del sistema puede comprender. Un **archivo ejecutable** es una serie de secciones de código que el cargador puede cargar en memoria y ejecutarse.

Los archivos pueden tener las características antes mencionadas de acuerdo con el tipo de información que contienen, además de tener una longitud y nombre; es importante que se conozcan, porque de esta forma se podrá tener mejor control de lo que se almacena y de cómo se almacena.

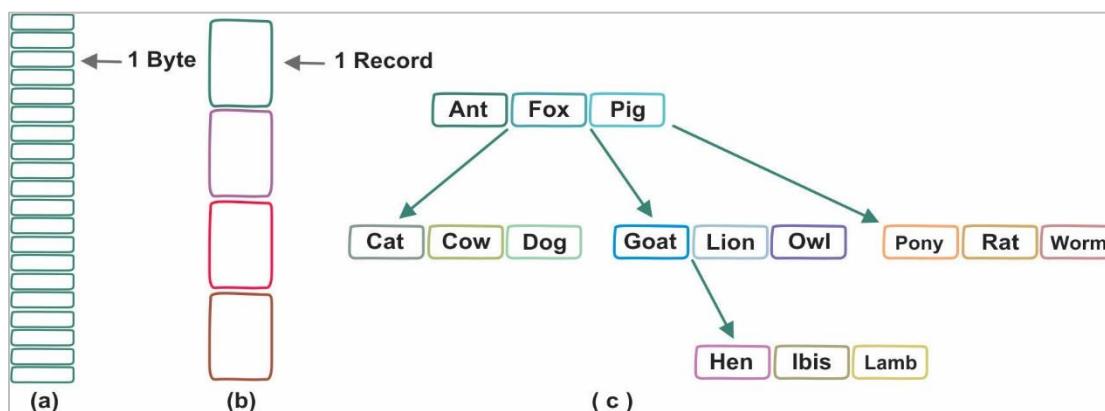
### 2.1.2. Jerarquía de datos

Es bastante común desear organizar los archivos de forma lógica, para ello se requiere agrupar los archivos de alguna forma dentro de esquemas flexibles determinados por cada usuario, para dar como resultado la **organización de la información lógica**.

Básicamente, lo que se requiere es contar con una jerarquía general de datos, comúnmente conocida como árbol de directorios; con este tipo de organización de información el usuario puede tener la cantidad de directorios que requiera para agrupar y organizar sus archivos en diferentes categorías. En la siguiente figura se especifica que existen directorios (A, B, C) pertenecientes de la raíz (root), los cuales son archivos de diferentes usuarios y éstos a su vez generan subdirectorios para los proyectos en los que se está trabajando.



Los usuarios y los programas de aplicación interactúan con el sistema de archivos por medio de órdenes de creación y eliminación de archivos, realizando operaciones sobre los archivos. A través de la estructura de archivos es posible tener un buen control y acceso a la información. El sistema es quien se encarga de realizar la entrada y salida de cada uno de esos directores, sin siquiera considerar la posibilidad de perderse a través de un buen mecanismo de E/S, como se pudo estudiar en la unidad 1, en el subtema Entrada /Salida.



Estructura de ficheros de tres tipos de jerarquía. (a) Sucesión de bytes. (b) sucesión de registros. (c) árbol. Tomada de Tanenbaum (2003, p. 382).



Los archivos o datos pueden estar estructurados de formas diferentes en la figura anterior se muestran tres posibilidades comunes. El ejemplo (a) es una sucesión no estructurada de bytes. Cualquier significado que se tenga sobre el fichero deberán atribuírselo los programadores en el nivel de usuario.

La flexibilidad que se ofrece es alta, mientras el sistema operativo ve solamente como secuencias de bytes, los programas de usuario pueden poner cualquier cosa que deseen los archivos y darles cualquier nombre que crean conveniente. En ese sentido, el sistema operativo se torna imparcial para que los usuarios puedan hacer cosas fuera de lo común, por ejemplo, ponerle cualquier nombre de archivo por más chusco o raro que éste parezca o guardar información en sus archivos que para otros usuarios carezca de sentido. Es decir, el sistema operativo no va a evaluar cómo o lo que se guarde siempre y cuando cumpla reglas como no incluir caracteres especiales, que son reservados exclusivamente para el manejo de la información, como son /, \* entre otros.

El ejemplo (b) de la figura anterior se muestra que un archivo es una secuencia de registros, esto se apoya en el concepto de que la operación de lectura devuelve un registro y que la operación de escritura sobrescribe o anexa un registro.

El ejemplo(c) indica una estructura de archivos. En esta organización, un archivo consiste en un árbol de registros, que no necesitan todos de la misma longitud, cada uno de éstos contiene un campo de llave que representa una posición fija dentro del registro. El árbol está ordenado según el campo llave, a fin de poder buscar rápidamente una llave particular.

La **estructura lógica** de los datos se organiza de acuerdo con la forma en que se accede a éstos. La estructura más simple de un directorio es una lista de entradas para cada archivo.



### 2.2. Tipo de archivos

Dependiendo del sistema operativo se establecen los tipos de archivos que se van a manejar, por estandarización, el mercado, en la actualidad, al diseñar sus sistemas, hace que se reconozcan los más comunes, a continuación, se presenta una selección y explicación de los mismos:

**Archivos de imagen:** son archivos creados por ciertos programas específicos y como su nombre lo dice su propósito es contener una imagen, en la actualidad este tipo de archivos se manejan muy estandarizados, no sólo para que sean reconocidos por el sistema operativo de una computadora, sino el del celular, cámaras fotográficas, de video y otros dispositivos electrónicos; otros dependen del programa en específico que los va a manipular y pueden usar extensiones, como: **ai**, reconocido por adobe ilustrator, dwg, manipulado por autocad; **png** (por sus siglas en inglés de Portable Network Graphics, en español gráficos portables de red), que es un formato gráfico muy completo, especialmente pensado para redes; **psd**, gráfico creado con el programa Adobe Photoshop; **bmp**, simplemente con mapas de bits para la imagen ec; **gif**, este formato fue diseñado por Compu Serve y debe su nombre a las siglas del inglés Graphic Interchange Format (formato gráfico de intercambio), es un archivo de imagen, con este formato se puede almacenar hasta 256 colores, por lo que generalmente son diseños sencillos, como rayas, tramados, dibujos animados simples, etc., y por su bajo peso y característica multiplataforma, se utiliza extensamente en Internet.

**Jpeg:** este formato es utilizado para almacenar y presentar fotos e imágenes sin movimiento. A diferencia de su similar gif, permite el manejo de un mayor número de colores. Jpeg es un estándar para el manejo de imágenes en Internet.

**Archivos de audio:** son aquellos que como su nombre lo dicen son para el manejo de todo lo relacionado con el audio, como pudieran ser canciones, videos, de igual manera guardan un tipo especial de extensión dependiendo del programa que los genere, pero también es una realidad que existe una estandarización para que los dispositivos electrónicos los reconozcan como ejemplo: **riff**, que identifica archivos que contienen audio en formato WAVE. Este formato, diseñado por Microsoft, es un estándar para el manejo de audio en la plataforma Windows y permite obtener una alta calidad en el almacenamiento y reproducción de audio,



aunque su principal desventaja es el elevado peso de los archivos que genera; **avi**, esta extensión identifica archivos que contienen vídeo digital. El formato AVI es el estándar utilizado en la plataforma Windows. Su principal desventaja es que genera archivos algo pesados; **mp3**, identifica archivos que contienen sonido en formato MPEG Audio Stream, Layer III; se trata de un formato no propietario. Este formato tiene la capacidad de almacenar audio con calidad similar a la de un disco compacto. Su ventaja radica en que se reduce considerablemente el peso de los archivos generados (normalmente la proporción es de 1/10). Por este motivo, mp3 se ha convertido en el formato más utilizado para la distribución de música en Internet.

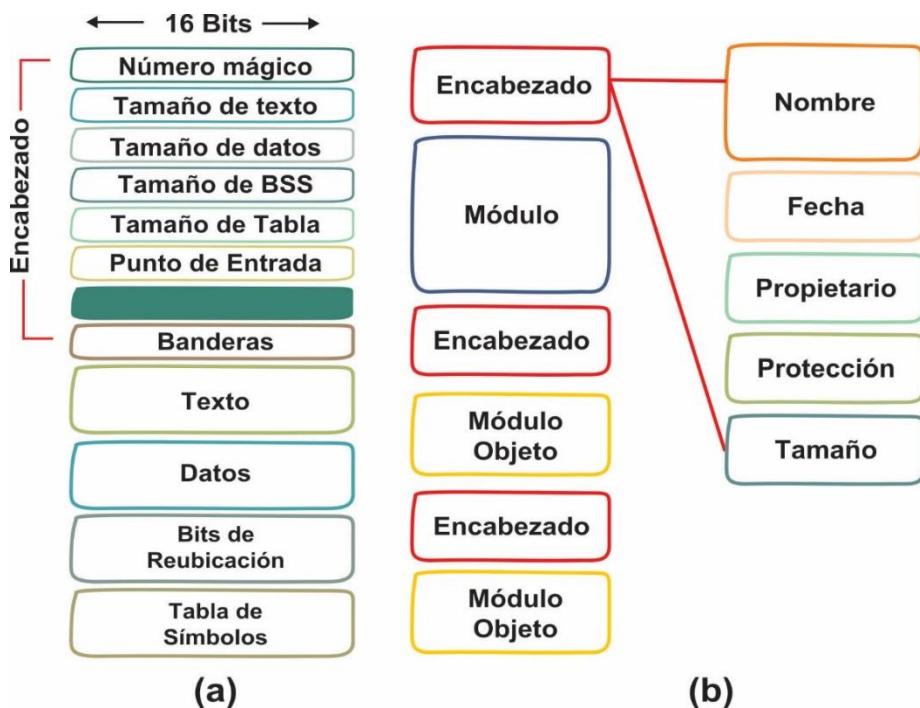
**Archivos de texto:** son aquellos que en su interior sólo almacenan texto, o sea caracteres y carecen de formato.

**Archivos ASCII:** tienen la ventaja de que pueden exhibirse e imprimirse tal como están, y se pueden editar con cualquier editor de textos. Además, si una gran cantidad de programas usan archivos ASCII como entradas y salidas, es fácil conectar la salida de un programa a la entrada de otro, como sucede con los conductos de Shell. Los archivos que no son ASCII se interpretan como archivos binarios.

### Archivos normales ASCII

Los **archivos binarios** son los que no se interpretan como archivo ASCII, si se obtiene una vista de los archivos binarios por medio de una impresión, se obtendrá un listado difícil de comprender lo que pareciera ser contenido basura; por lo general estos archivos están formados por su estructura propiamente interna.

En la siguiente figura se puede apreciar la ejecución de un archivo binario simple; el ejemplo fue tomado de las primeras versiones de UNIX. En la secuencia de bytes representado por el archivo, se observa que el sistema operativo únicamente ejecutará un archivo si tiene el formato que se reconoce por el sistema operativo.



Ejecución de un archivo binario simple. Tomada de Tanenbaum (2003, p. 406).

En el ejemplo (a) se cuenta con cinco secciones:

- **Encabezado.** Es quien inicia e identifica el archivo como ejecutable y valido para el sistema operativo.
- **Texto.** Se carga en memoria y utiliza la reubicación de bits para establecer su ubicación dentro del sistema.
- **Datos.** Al igual que en la sección texto se carga en memoria y utiliza la reubicación de bits para establecer su ubicación dentro del sistema.
- **Bits de reubicación.** Se utilizan para la reubicación de los encabezados de texto y datos.
- **Tabla de símbolos.** Es utilizada para la depuración.

Tomando el segundo ejemplo, (b), se tiene que el archivo binario se considera como un archivo, el cual consiste en la colección de procedimientos de las bibliotecas o módulos que no están enlazados. Cada módulo se representa con un encabezado que es identificado como el nombre, fecha de creación, propietario, código de protección y tamaño. De la misma forma que el archivo ejecutable, los encabezados de módulo están llenos de números binarios.



Como te diste cuenta existen muchísimos tipos de archivos, aquí sólo se mencionaron algunos, pero seguramente has encontrado en celulares, cámaras fotográficas y de video, reproductores de música, incluso en una computadora, que para funcionar todos éstos cuentan con un sistema operativo, y según el sistema a diseñar y la utilidad que va a tener debes pensar qué tipos de archivos y cómo los va a manejar el sistema que estás creando.

### 2.2.1. Real

El **archivo real** en Windows es el que es considerado como un objeto; está formado por programas, datos o cualquier tipo de información que contenga uno o varios elementos dentro del sistema operativo. Un archivo real se muestra por el espacio que ocupa dentro de un disco duro o el mismo sistema de almacenamiento de archivos administrado por el sistema operativo.

Es decir, se considera cualquier archivo que ocupe un espacio en una computadora y que se coloque ahí o que el mismo sistema lo haya hecho, sin importar si ocupa poco o mucho espacio. Puede ser colocado en cualquier parte del disco duro o directorio que se desee. La siguiente imagen muestra un grupo de cinco archivos reales, en este caso tres de texto, dos de tipo HTML y una carpeta que contiene más archivos.



Grupo de archivos reales y una carpeta que contiene más archivos

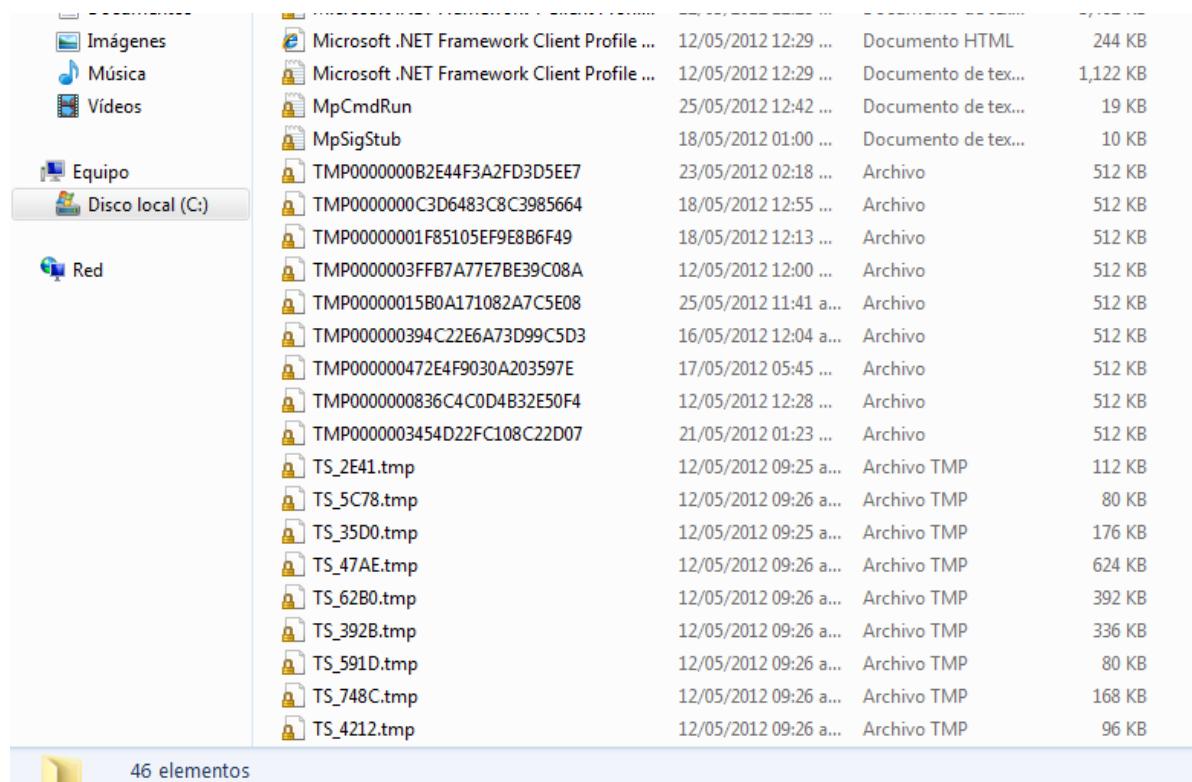
En resumen, todo aquello que se guarde en una computadora el sistema operativo lo interpreta como un archivo real.



### 2.2.2. Virtual

En contra parte de un archivo real, el virtual no ocupa disco duro de nuestra computadora, aunque si existe y tiene un tamaño específico, éste es temporal, y por lo tanto sólo se almacena en el espacio designado a la memoria **RAM**.

Mucha de las veces es creada por el sistema mismo, por ejemplo, cuando se abre un archivo de texto y se manipula, el sistema de manera interna crea una copia y coloca el símbolo ~ al principio del mismo para distinguir que es temporal, y que es el archivo que está sufriendo las modificaciones mientras no se realice el proceso de guardado del archivo real. Los nuevos sistemas operativos ocultan este tipo de archivos siendo invisibles para los usuarios. Otros archivos que son considerados virtuales son aquellos que de igual forma se alojan en memoria **RAM** y que son generados el sistema mismo, claro ejemplo de ello se puede encontrar en el directorio **temp** del sistema Windows.



Nombre	Fecha	Tipo	Tamaño
Microsoft .NET Framework Client Profile ...	12/05/2012 12:29 ...	Documento HTML	244 KB
Microsoft .NET Framework Client Profile ...	12/05/2012 12:29 ...	Documento de tex...	1,122 KB
MpCmdRun	25/05/2012 12:42 ...	Documento de tex...	19 KB
MpSigStub	18/05/2012 01:00 ...	Documento de tex...	10 KB
TMP0000000B2E44F3A2FD3D5EE7	23/05/2012 02:18 ...	Archivo	512 KB
TMP0000000C3D6483C8C3985664	18/05/2012 12:55 ...	Archivo	512 KB
TMP00000001F85105EF9E8B6F49	18/05/2012 12:13 ...	Archivo	512 KB
TMP00000003FFB7A77E7BE39C08A	12/05/2012 12:00 ...	Archivo	512 KB
TMP00000015B0A171082A7C5E08	25/05/2012 11:41 a...	Archivo	512 KB
TMP000000394C22E6A73D99C5D3	16/05/2012 12:04 a...	Archivo	512 KB
TMP000000472E4F9030A203597E	17/05/2012 05:45 ...	Archivo	512 KB
TMP0000000836C4C0D4B32E50F4	12/05/2012 12:28 ...	Archivo	512 KB
TMP0000003454D22FC108C22D07	21/05/2012 01:23 ...	Archivo	512 KB
TS_2E41.tmp	12/05/2012 09:25 a...	Archivo TMP	112 KB
TS_5C78.tmp	12/05/2012 09:26 a...	Archivo TMP	80 KB
TS_35D0.tmp	12/05/2012 09:25 a...	Archivo TMP	176 KB
TS_47AE.tmp	12/05/2012 09:26 a...	Archivo TMP	624 KB
TS_62B0.tmp	12/05/2012 09:26 a...	Archivo TMP	392 KB
TS_392B.tmp	12/05/2012 09:26 a...	Archivo TMP	336 KB
TS_591D.tmp	12/05/2012 09:26 a...	Archivo TMP	80 KB
TS_748C.tmp	12/05/2012 09:26 a...	Archivo TMP	168 KB
TS_4212.tmp	12/05/2012 09:26 a...	Archivo TMP	96 KB

Directorio temp de sistema operativo Windows

En conclusión, un archivo virtual existe, pero alojado en la memoria **RAM**; es para uso interno del sistema operativo, y al estar alojado en memoria **RAM**, al apagar el sistema, se pierde.



### 2.2.3. Componentes

Los componentes principales para un archivo se forman por un nombre y datos. Además, la mayoría, o todos los sistemas operativos relacionan otra información a cada archivo contenido, tales como los componentes de fecha y hora de su creación, además del tamaño del archivo. Por lo general, un componente es en realidad el atributo del archivo, el cual varía bastante de un sistema operativo a otro, pues la gestión de la información la asocian a su sistema de archivos.

La figura siguiente muestra un esquema básico y general sobre algunos atributos, sin dejar de pensar que pueden existir muchas más. La mayoría de los sistemas operativos puede que no integren la mayoría de los atributos que se muestran, pero todos están presentes en algún sistema.

Atributo	Significado
Protección	Quién puede tener acceso al archivo y cómo
Contraseña	Clave necesaria para tener acceso al archivo
Creador	ID de la persona que creó el archivo
Dueño	Propietario actual
Indicador de sólo lectura	0 para leer/escribir; 1 para sólo lectura
Indicador de oculto	0 para normal; 1 para no mostrar en listados
Indicador de sistema	0 para archivos normales; 1 para archivo de sistema
Indicador de archivado	0 para ya respaldado; 1 para debe respaldarse
Indicador de ASCII/binario	0 para archivo ASCII; 1 para archivo binario
Indicador de acceso aleatorio	0 para sólo acceso secuencial; 1 para acceso aleatorio
Indicador de temporal	0 para normal; 1 para borrarlo al terminar el proceso
Indicadores de bloqueo	0 para sin bloqueo; distinto de cero si está bloqueado
Longitud de registro	Número de bytes en un registro
Posición de clave	Distancia a la clave, dentro de cada registro
Longitud de clave	Número de bytes en el campo clave
Hora de creación	Fecha y hora en que se creó el archivo
Hora de último acceso	Fecha y hora en que se tuvo acceso por última vez al archivo
Hora de último cambio	Fecha y hora en que se modificó por última vez el archivo
Tamaño actual	Número de bytes en el archivo
Tamaño máximo	Número de bytes que puede alcanzar el archivo

Algunos de los componentes de archivos comúnmente utilizados  
Tomada de Tanenbaum (2003, p. 387).



En el listado que se muestra en la tabla anterior, los primeros cuatro atributos se refieren a la protección del archivo, e indican quién puede tener acceso al mismo y quienes tienen el acceso restringido. En una gran mayoría de sistemas, para permitir el acceso al sistema o algún archivo el usuario debe presentar una contraseña para poder validar que tiene el permiso de ingreso, en ese caso la contraseña que se valida por el sistema deberá ser el atributo.

Los indicadores son bits o campos cortos que controlan y habilitan alguna propiedad específica, los archivos ocultos, por ejemplo, no aparecen en los listados de todos los ficheros. El indicador de archivado es un bit que especifica si el archivo ya se respaldó o no. El programa de respaldo sabe qué ficheros deben respaldarse. El indicador temporal permite marcar un fichero para que se elimine de forma automática cuando termine el proceso que lo creó (Tanenbaum, 2003).

Dentro de los campos de longitud de cada registro, la posición de la clave y longitud de la clave se presentan en los ficheros cuando los registros puedan ocultarse empleando una clave. Dichos campos proporcionan la información necesaria para hallar las claves.

El tamaño de un archivo indica qué tanto espacio llega a ocupar dentro del disco duro, una gran parte de los sistemas operativos antiguos exigían que fuera especificado el tamaño del archivo al momento de ser creado con la intención de poder administrar la cantidad máxima de espacio para su almacenamiento. Los sistemas operativos de estaciones de trabajo y ordenadores personales son bastante inteligentes para prescindir de esa información.

### 2.3. Organización del sistema de archivos, lógica y física

El sistema de archivos no es más que el encargado de organizar la información dentro de los dispositivos de almacenamiento de manera que el sistema operativo pueda entender esa organización.

En un planteamiento en un nivel un poco más físico, un sistema de archivos es una forma de organizar los archivos dentro de una partición.

Se define al sistema de gestión de archivos (Carretero, 2001) como aquel sistema software que proporciona a los usuarios unos servicios relativos al empleo de archivos.



El sistema de archivos es por tanto un intermediario entre el tratamiento lógico de los archivos, que es el que permite que los usuarios puedan utilizarlos, y el tratamiento físico, que es el que se encarga de organizarlos en disco. Los sistemas operativos implementan diferentes métodos de acceso a los archivos. De acuerdo con la organización lógica, los más habituales son:

- **Organización secuencial.** Es considerada la más sencilla de implementar, porque los registros se almacenan y se pueden recuperar en serie. Para localizar un registro específico, se busca en el archivo desde el inicio hasta encontrar el registro solicitado.
- **Organización directa.** Son archivos de acceso directo cuyo almacenamiento sólo se pueden implementar en dispositivos de almacenamiento de acceso directo, con la diferencia de la organización secuencial, la organización directa permite acceder a cualquier registro en cualquier orden, sin la necesidad de iniciar la búsqueda desde el principio del archivo.
- **Organización secuencial indexada.** Este tipo de organización de archivos es una combinación de los dos tipos de organizaciones anteriores, se crea y mantiene a través de un paquete de software de método (Flynn, 2001).

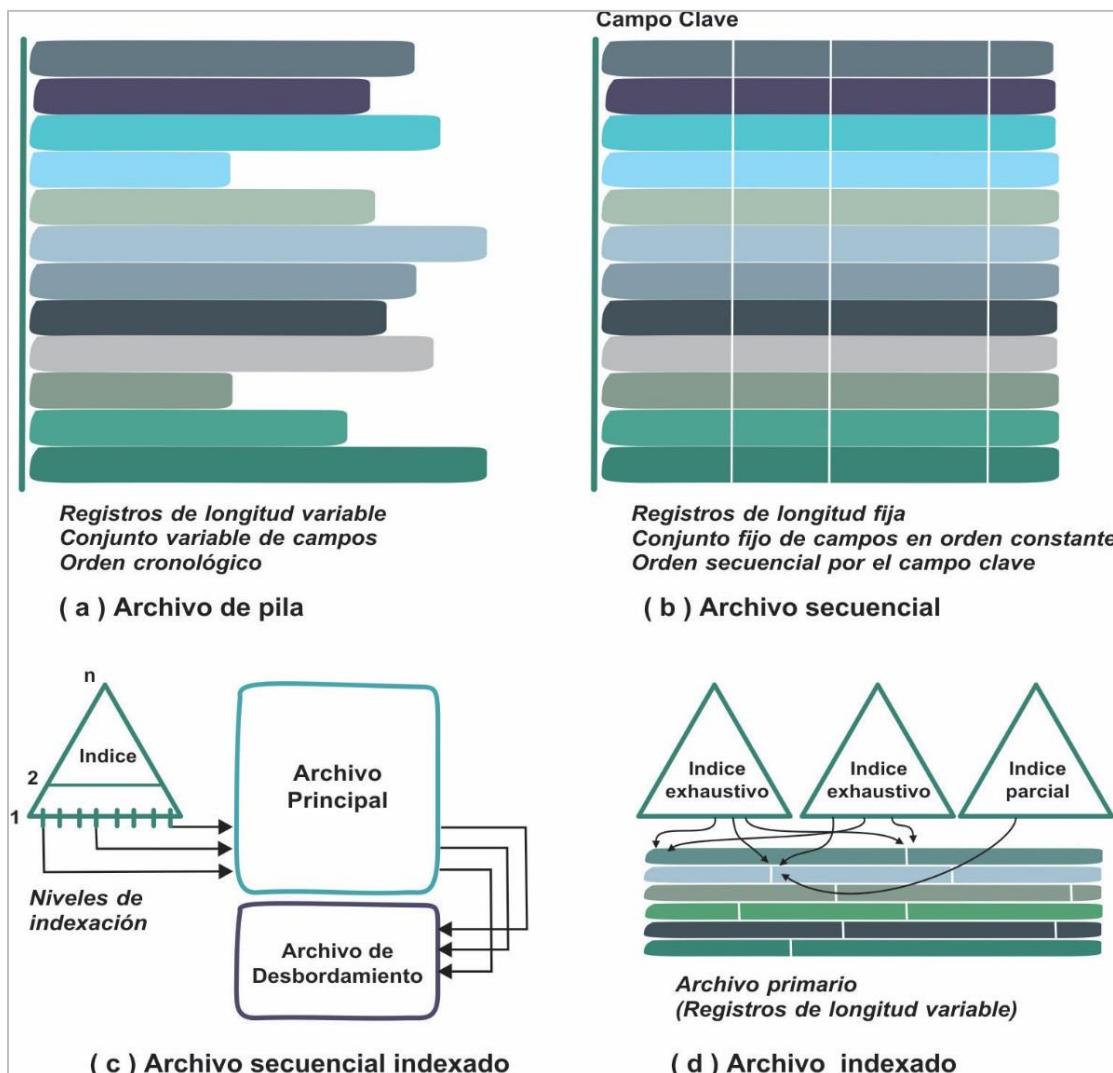
Para lograr la mejor selección de estas opciones, por lo general se consideran estas características:

- **Volatilidad de los datos.** Frecuencia con la cual se efectúan adiciones y eliminaciones.
- **Actividad del archivo.** Porcentaje de registros procesados durante una ejecución.
- **Tamaño del archivo.** Proporción de espacio que ocupa el archivo.
- **Tiempo de respuesta.** “Cantidad de tiempo que el usuario acepta esperar antes que la operación solicitada se complete” (Flynn, 2001). Es recomendable consultar el tema de organización de los archivos en el texto de Flynn (2001), donde encontrarás los tipos de organización de archivos secuencial directa e indexada en la página 190.

La organización física del sistema de archivos en memoria secundaria depende de la estrategia de los diferentes tipos de organización que se mencionaron antes, y se representan en la siguiente figura, que muestra que los archivos de pila contienen longitud variable y se van colocando conforme el tiempo en el que fueron creados, cada rayita de color representa un archivo, en el inciso (a). El inciso (b) muestra registros de longitud fija y si se observa cada rayita de color representa un archivo, los cuales son del mismo tamaño



y se van colocando de manera secuencial, es decir, uno después de otro. En el inciso (c) se muestran los archivos con un secuencial indexado, es decir, contiene un índice de cómo están los archivos; es secuencial porque se coloca un archivo detrás de otro y el desbordamiento es cuando es más grande de tamaño. Por último, el inciso (d) muestra archivos indexados colocados en desorden y los índices se encargan de saber dónde está cada archivo.



Diferentes técnicas de organización de archivos. Tomada de Stallings (2005).



Otros de los criterios importantes a considerarse durante la elección de una organización de archivos son:

- Acceso directo para la recuperación rápida de la información.
- Facilidad de actualización para colaborar a mantener la información.
- Economía para minimizar el costo de almacenaje.
- Facilidad de mantenimiento simple para minimizar la probabilidad de errores.
- Confianza de asegurar los datos.

En la organización de archivos física, los datos son modificados en su acceso contigo físico, dependiendo del tipo de dispositivo de almacenamiento secundario, los registros pueden ser de tamaño fijo o variable y se pueden organizar de distintas formas para construir archivos físicos.

En la actualidad existen diferentes medios de almacenamiento físico de los datos en el mercado; esto es algo que está en constante evolución por el simple hecho que día a día los dispositivos tienen más capacidad y ocupan menor tamaño, algunos de los más comunes para grandes cantidades de información son:

**Cinta magnética:** contiene en su interior una banda plástica, como su nombre lo dice, magnética, se encuentra dentro de un casete por consecuencia los datos se tienen que almacenar de forma secuencial; un ejemplo de éstas son las que se utilizaban para video, conocidas como VHS, tiene gran desventaja que para acceder a la información tiene que hacerse el recorrido por la cinta, pero de su ventaja son las grandes capacidades de información que puede almacenar.

**Disco magnético:** son los más utilizados en el mercado y son los que contienen cualquier computadora personal. Se compone de discos o platos que guardan la información en ambas caras del plato y se accede a través de ellos por medio de una aguja lectora, no necesariamente su información debe ser secuencial, puede también ser indexada, ya que la aguja se desplaza hasta donde lo necesite. Como te has dado cuenta almacenar la información no es el problema, el problema es cuando se borra, qué sucede con estos espacios libres para el control de estos espacios; existen muchas técnicas que pueden llegar a consistir en el reacomodo de la información o generar nuevos índices para utilizar estos espacios.

### 2.3.1. Operaciones con archivos



Comúnmente los archivos están hechos para almacenar la información que se necesita recuperar luego de realizar algún proceso de datos. La mayoría de los sistemas administradores de archivos brindan algunas operaciones de almacenamiento y recuperación, en siguiente listado se muestran las llamadas más comunes relacionadas con archivos:

Create	Read	SetAttributtes
Delete	Append	Rename
Open	Seek	
Close	GetAttributes	

Todas estas operaciones las describe Tanenbaum (2003), en su obra *Sistemas operativos modernos* en la página 387:

- **Create.** Generalmente esta llamada genera el archivo sin dato alguno, su propósito es informar que va a existir un archivo y sólo establece algunos de su componentes o atributos (en tema anterior se mostraron los diferentes tipos de atributos).
- **Delete.** Esta llamada es muy común utilizarla cuando existe un archivo dentro del sistema y ya no es necesario su existencia para liberar espacio en disco.
- **Open.** La utilidad de esta llamada es habilitar al sistema que se le otorguen los atributos y la lista de direcciones de disco, y los coloque en la memoria principal a fin de agilizar el acceso en llamadas posteriores.
- **Close.** Esta llamada es útil para cuando todos los procesos de los archivos estén concluidos; sus atributos y las direcciones de disco no sean de utilidad, por lo general es necesario cerrar el archivo para liberar espacio correspondiente en las tablas internas.
- **Read.** Llamada útil para leer los datos del archivo, por lo general los bytes provienen de la posición actual. El invocador debe especificar cuántos datos se necesitan y también un buffer para colocarlos.
- **Write.** Se utiliza esta llamada para modificar los datos en los archivos, en su posición actual. Si esta posición está en el final del archivo, el tamaño aumenta. Se sobrescriben cuando la posición del archivo esté a la mitad de su posición y son reemplazados por los existentes.



- **Append.** Esta llamada realiza llamadas de write pero con la restricción de agregar los datos al final del archivo. Los sistemas que ofrecen un juego mínimo de llamadas al sistema generalmente no cuentan con append.
- **Seek.** En la organización de acceso aleatorio de los archivos, necesita un método para especificar el lugar del que deben tomarse los datos. Un enfoque común es tener una llamada al sistema, Seek, que ajustará a la posición actual del archivo una vez que localiza su ubicación se podrán leer los datos de esa posición y escribir en ella.
- **GetAttributes.** Comúnmente los procesos requieren leer los atributos de los archivos para realizar alguna llamada o proceso, realiza un examen de algunos de sus atributos para saber los tiempos de modificación.
- **Set Attributes.** Algunos de los atributos de los archivos pueden ser establecidos por el usuario y modificarse después de que se creó el archivo.
- **Rename.** Frecuentemente los usuarios requieren de modificar el atributo del nombre al archivo que existe, esta llamada permite hacerlo.

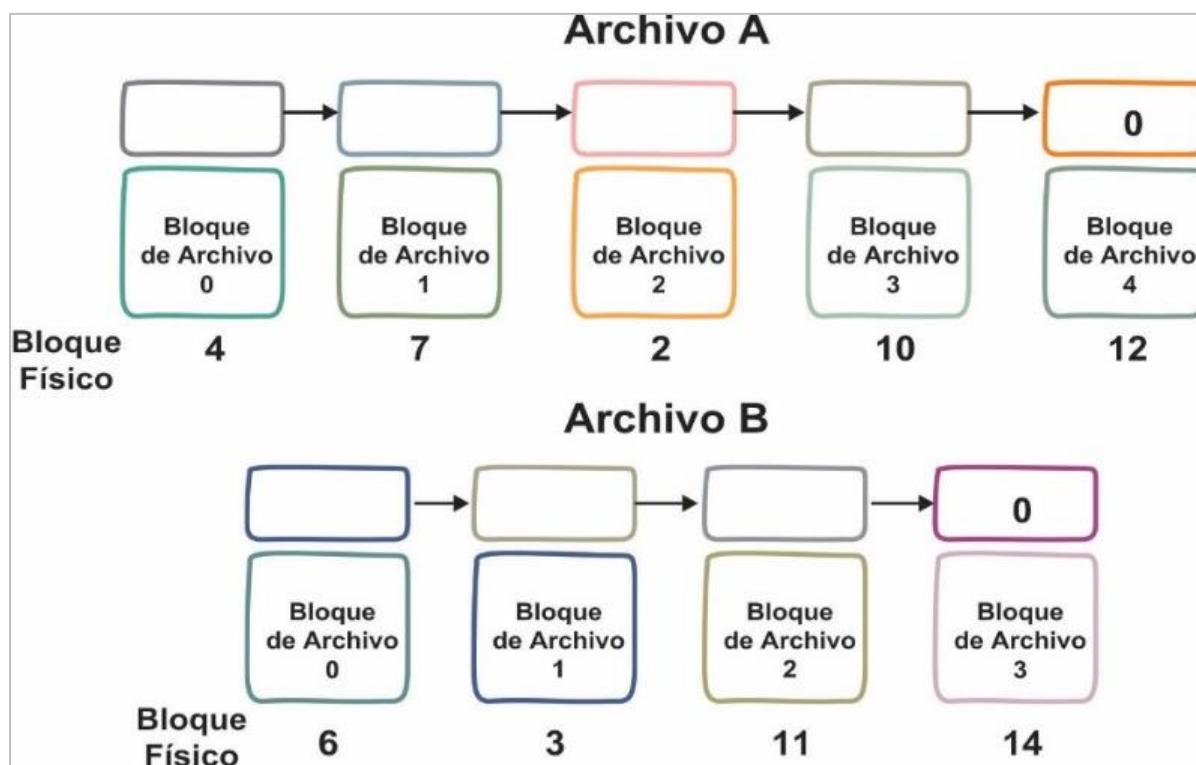
### 2.3.2. Implementación de archivos

Possiblemente para los usuarios el punto más importante de la implementación del almacenaje de los archivos es poder llevar un control acerca de qué bloques de disco corresponden a qué fichero. Se contemplan varios métodos en los distintos sistemas operativos, los cuales son:

- **Asignación contigua.** Es considerado el esquema más simple, pues almacena cada archivo como un bloque secuencial de datos en el disco, por ejemplo, en un disco con bloques de 1 byte a un archivo de 50 bytes se le asignarán 50 bloques consecutivos (Tanenbaum, 2003). Una de las principales ventajas, es la sencillez con que sabe dónde están los bloques de un archivo y el rendimiento, es bastante bueno, pues es posible leer todo el archivo del disco en una sola operación. Así como existen ventajas de este método de implantación, también cuenta con algunas desventajas, como que no es muy factible si no se conoce el tamaño máximo del archivo en el momento en que se crea éste, ya que el sistema operativo no se dará cuenta: cuánto espacio en disco debe reservar para este archivo. Y en la fragmentación del disco se desperdicia espacio que de otra forma podría haberse aprovechado.

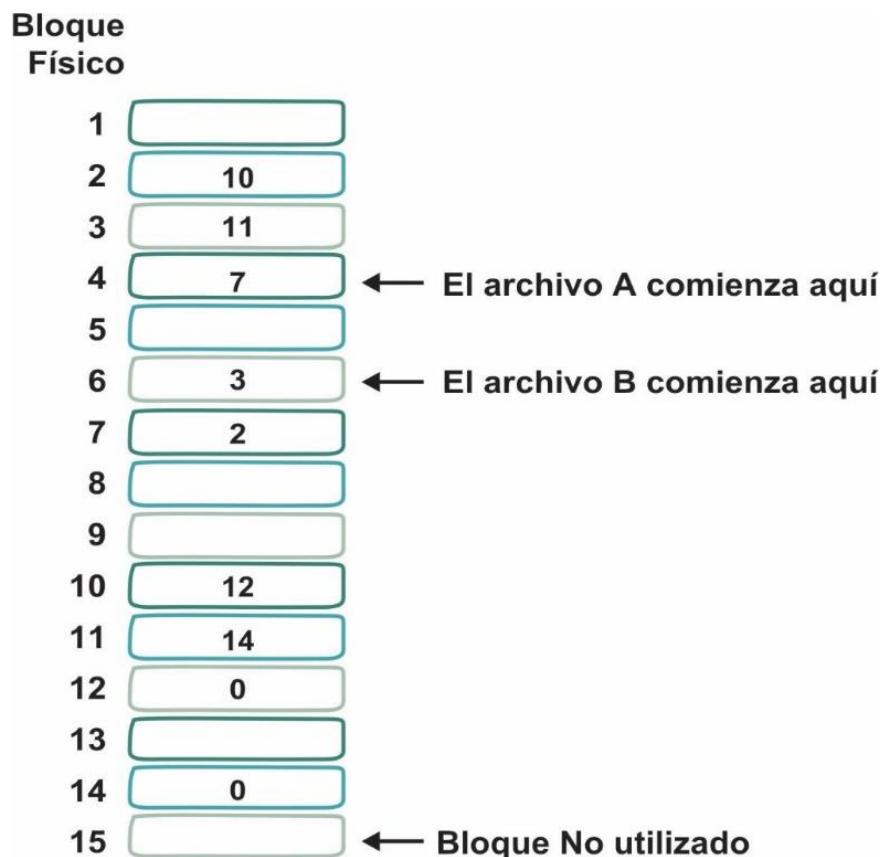


- **Asignación por lista enlazada.** El segundo método para el almacenamiento de archivos es guardar cada uno como una lista enlazada de bloques de disco, en la siguiente figura se indica que la primera palabra de cada bloque se emplea como línea enlazada al siguiente bloque (Tanenbaum, 2003). Este método tiene la posibilidad de utilizar todos los bloques y no pierde espacio por la fragmentación de disco.



Asignación por lista enlazada de bloques de disco. Tomada de Tanenbaum (2003, p. 403).

- **Asignación por lista enlazada utilizando índices.** Al utilizar esta implementación todo el bloque estará disponible por completo para los datos, y el acceso directo es más sencillo a pesar de que se tiene que seguir la cadena para encontrar una distancia dada dentro de un archivo, la cadena está por completo en la memoria y puede seguirse sin tener que consultar el disco. Su principal desventaja es que toda la tabla debe estar en la memoria para que pueda funcionar; en discos grandes se vería afectado, pues entre mayor capacidad del disco, menor será el rendimiento de búsqueda. La siguiente figura muestra el bloque de asignación por lista, donde se puede iniciar en el bloque 4 y seguir la cadena hasta el final del bloque y lo mismo para el bloque 6, iniciar desde ahí hasta seguir la cadena final.



Asignación por lista enlazada utilizando índices

Tomado de Tanenbaum (2003, p. 404).

- **Nodos i (nodo índice).** Este método de implementación es útil para saber cuáles bloques pertenecen a cuál archivo; consiste en asociar cada archivo a una pequeña tabla, la cual contiene los atributos y direcciones en disco de los bloques del archivo. En la siguiente figura se muestra un ejemplo simple, en donde la posibilidad de encontrar todos los bloques del archivo. Su principal ventaja, respecto al anterior es que los nodos índices emplean una tabla en la memoria, porque sólo debe estar en memoria mientras el archivo correspondiente está abierto. Otra ventaja de este método es el rendimiento que ofrece, sólo es necesario reservar cantidad de espacio para los archivos abiertos.

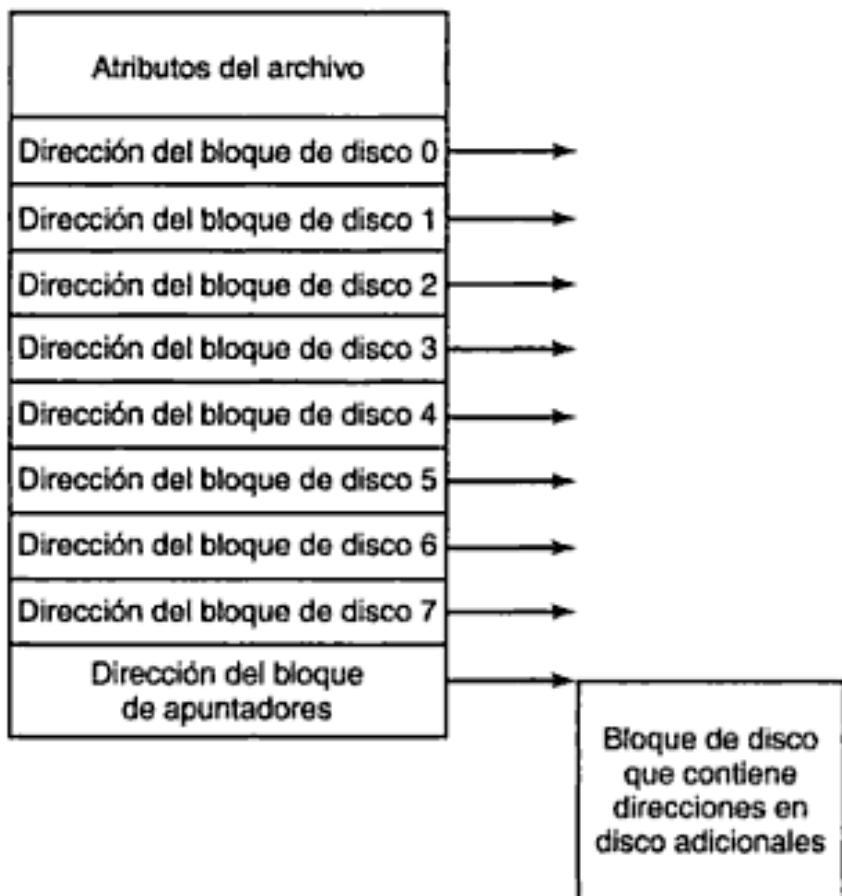


Tabla de método de implementación de archivos por medio de nodos índice  
Tomada de Tanenbaum (2003).

Una de las desventajas de los nodos índices es que fija el espacio de las direcciones en disco. Pero mientras el archivo crece puede reservar la última dirección de disco no para un bloque de datos, sino para la dirección de un bloque que contiene más direcciones de bloques de disco.



### Cierre de la unidad

Has concluido el estudio de la segunda unidad. A lo largo de ésta se vieron conceptos básicos sobre el administrador de sistemas de archivos, sus conceptos y la jerarquía de datos, los tipos de archivos reales y virtuales; así como sus componentes, además de la organización del sistema de archivos de manera lógica y física, las operaciones que se pueden realizar con los mismos y cómo se implementan.

Es recomendable que revises nuevamente la unidad en caso de que los temas que se acaban de mencionar no te sean familiares o no los recuerdes, de lo contrario ya estás preparado (a) para seguir con la Unidad 3. Seguridad y protección, en donde se abordará lo relacionado con la seguridad y protección del entorno, el concepto y objetivo de la protección. Todo ello con el fin de obtener un prototipo final al concluir las cuatro unidades de Programación de sistemas operativos.

### Para saber más

Si deseas saber más acerca de los sistemas operativos, revisa el siguiente sitio donde encontrarás un curso completo universitario de sistemas operativos convencionales y distribuidos con posibilidades de descarga.

Martínez, D. L. (2010, 22 de mayo). *Sistemas de archivos*.

[http://sistop.gwolf.org/html/biblio/Sistemas\\_Operativos - Luis La Red Martinez.pdf](http://sistop.gwolf.org/html/biblio/Sistemas_Operativos - Luis La Red Martinez.pdf)



### Fuentes de consulta

#### Básica

Candela, S. y García, C. (2007). *Fundamentos de sistemas operativos. Teoría y ejercicios resueltos*. España: Paraninfo.

Duran, L. (2000). *Referencia básica sistemas operativos*. España: Marcombo.

Flynn, I. (2001). *Sistemas operativos*. México: Thomson.

Morera, J. y Pérez-Campanero, J. (2002). *Conceptos de sistemas operativos*. Madrid: Comillas

Niño, J. (2011). *Sistemas operativos monoexpuesto*. Madrid: Editex.

Ortiz, H. (2005) *Sistemas operativos modernos*. Medellín: Universidad de Medellín.

Silberschatz, A. Korth, H. F. y Sudarshan, S. (2006). *Fundamentos de sistemas operativos*. Madrid: McGraw-Hill.

Stallings, W. (2005). *Sistemas operativos: aspectos internos y principios de diseño*. Madrid: Pearson-Prentice Hall.

Tanenbaum, A. (2003). *Sistemas operativos modernos*. México: Pearson Educación.  
<https://tinyurl.com/yytqj8hx>

#### Bibliografía complementaria

Doménech Gómez, S., & Huguet Soriano, O. V. (s.f.). *Linux. Sistema de ficheros*. Mural de la Universidad Valenciana. <http://mural.uv.es/oshuso/index.html>

Silberschatz, A., Galvin, P. B., & Gagne, G. (2006). *Fundamentos de sistemas operativos*. 7<sup>a</sup>. México: McGraw Hill Interamericana de España.