

# SISTEMAS DE ARCHIVOS EN LINUX

Decidí escribir este artículo ,ya que la mayoría de usuarios Linux utilizan sus particiones con ext3. Este artículo no intenta decir que sistema de archivo es mejor, solo mostrara sus ventajas y las desventajas. Así como diversos ejemplos de su utilización.

## ¿Qué es un sistema de archivos?

Los sistemas de archivos más comunes utilizan dispositivos de almacenamiento de datos (Disco Duros, CDS, Floppys, USB Flash , etc..) que permiten el acceso a los datos como una cadena de bloques de un mismo tamaño, a veces llamados sectores, usualmente de 512 bytes de longitud. El software del sistema de archivos es responsable de la organización de estos sectores en archivos y directorios y mantiene un registro de qué sectores pertenecen a qué archivos y cuáles no han sido utilizados.

un sistema de archivos tiene directorios que asocian nombres de archivos con archivos, usualmente conectando el nombre de archivo a un índice en una tabla de asignación archivos de algún tipo, como FAT en sistemas de archivos MS-DOS o los inodos de los sistemas Unix. La estructura de directorios puede ser plana o jerárquica (ramificada o "en árbol"). En algunos sistemas de archivos los nombres de archivos son estructurados, con sintaxis especiales para extensiones de archivos y números de versión. En otros, los nombres de archivos son simplemente cadenas de texto y los metadatos de cada archivo son alojados separadamente.

En sistemas de archivos jerárquicos, en lo usual, se declara la ubicación precisa de un archivo con una cadena de texto llamada "ruta". La nomenclatura para rutas varía ligeramente de sistema en sistema, pero mantienen por lo general una misma estructura. Una ruta viene dada por una sucesión de nombres de directorios y subdirectorios, ordenados

jerárquicamente de izquierda a derecha y separados por algún carácter especial que suele ser una barra (/) o barra invertida (\) y puede terminar en el nombre de un archivo presente en la última rama de directorios especificada.

**Por ejemplo, en un sistema Unix (También Linux) la ruta a la canción preferida del usuario "armando" sería algo como:**

```
/home/armando/multimedia/obizienk.mp3
```

**Un ejemplo análogo en un sistema de archivos Windows se vería como:**

```
C:\Mis documentos\Mi música\obizienk.mp3
```

## Sistema de Archivos de Disco

Un sistema de archivo de disco está diseñado para el almacenamiento de archivos en una unidad de disco, que puede estar conectada directa o indirectamente a la computadora.

Ejemplos de sistemas de archivos de disco:

- \* EFS
- \* EXT2
- \* EXT3
- \* FAT (sistemas de archivos de DOS y Windows)
- \* UMSDOS
- \* FFS
- \* Fossil
- \* HFS (para Mac OS)
- \* HPFS
- \* ISO 9660 (sistema de archivos de solo lectura para CD-ROM)
- \* JFS
- \* kfs
- \* MFS (para Mac OS)
- \* Minix

\* NTFS (sistemas de archivos de Windows NT - XP)

\* OFS

\* ReiserFS

\* Reiser4

\* UDF (usado en DVD y en algunos CD-ROM)

\* UFS

\* XFS

El ext2 permite asignar un porcentaje del disco duro que se reserva para el uso de usuarios con un "uid" y "gid" específicos (Quotas).

Más información:

<http://e2fsprogs.sourceforge.net/ext2.html>

## EXT3

En este artículo nos ocuparemos de 6 sistemas de archivos de Linux (EXT2, EXT3, ReiserFS, Reiser 4, XFS y JFS).

Decidí escribir acerca de estos sistemas de archivos, ya que el kernel 2.4 y 2.6 tienen soporte para estos. También algunas distribuciones como Debian te permiten seleccionar cualquiera de los 4 sistemas de archivos, desde la instalación.

Muy bien, empecemos..

## EXT2

EXT2 (second extended filesystem o "segundo sistema de archivos extendido") fue el sistema de archivos estándar en el sistema operativo GNU/Linux por varios años y continúa siendo ampliamente utilizado. Fue diseñado originalmente por Rémy Card. La principal desventaja de ext2 es que no implementa el registro por diario o bitácora (en inglés Journaling), que sí soporta su sucesor ext3 u otros sistemas de archivos como ReiserFS.

El ext2 tiene un límite máximo de 4GB de archivo, pero no limita el tamaño máximo de la partición a 4GB, como es el caso de la FAT.

El ext2 tiene soporte para detección de un sistema de archivos desmontado incorrectamente cuando el sistema se apaga de forma errónea, y capacidad para autorecuperarlo en caso de fallo accidental.

El ext2 mantiene información de la última vez que se montó y se accedió al volumen (sistema de archivos), así como del número de veces que se ha montado dicho volumen desde la última comprobación automática, así como la fecha en la que se comprobó su integridad por última vez.

Ext3 (third extended filesystem o "tercer sistema de archivos extendido") es compatible con Ext2, en realidad es Ext2 con un fichero adicional de registro. Ext3 es una capa adicional sobre Ext2 que mantiene un fichero de registro (por defecto en el directorio /jfs). Debido a que está integrado en el Ext2, sufre algunas de las limitaciones de dicho sistema, y no explota las posibilidades de los sistemas de journaling puros. Por ejemplo, todavía usa asignación basada en bloques y búsqueda secuencial de directorios, aunque se está trabajando en esta área para mejorarla.

Sus mayores ventajas son:

\* Ext3 mantiene la consistencia tanto en la meta-información como en los datos de los ficheros. A diferencia de los demás sistemas de journaling mencionados, la consistencia de los datos también está asegurada.

\* Las particiones Ext3 no tienen una estructura de ficheros diferentes a los de Ext2, por lo que no sólo se puede pasar de Ext2 a Ext3, sino que lo opuesto también funciona, útil sobre todo si en algún caso el registro se corrompe accidentalmente, por ejemplo debido a sectores malos del disco.

Ext3 reserva uno de los i-nodos especiales de Ext2 para el registro, pero los datos del mismo pueden estar en cualquier conjunto de bloques, y en cualquier sistema de ficheros. Incluso se puede compartir el registro entre sistemas distintos.

Tres tipos de bloques de datos son grabados en el registro:

1. Meta-información: contiene el bloque de meta-

información que está siendo actualizado por la transacción. Cada cambio en el sistema de ficheros, por pequeño que sea, es escrito en el registro. Sin embargo es relativamente barato ya que varias operaciones de E/S (Entrada/Salida) pueden ser agrupadas en conjuntos más grandes y pueden ser escritas directamente desde el sistema page-cache usando la estructura `buffer_head`.

2. Bloques descriptores: Estos bloques describen a otros bloques del registro para que luego puedan ser copiados al sistema principal. Los cambios en estos bloques son siempre escritos antes que los de meta-información.

**Bloques cabeceras:** Describen la cabecera y cola del registro más un número de secuencia para garantizar el orden de escritura durante la recuperación del sistema de ficheros.

La única diferencia entre EXT2 y EXT3 es el registro por diario. Un sistema de archivos EXT3 puede ser montado y usado como un sistema de archivos EXT2

Más información:

<http://e2fsprogs.sourceforge.net/ext2.html>

## ReiserFS

ReiserFS es un sistema de archivos de propósito general, diseñado e implementado por un equipo de la empresa Namesys, liderado por Hans Reiser.

A partir de la versión 2.4.1 del núcleo de Linux, ReiserFS se convirtió en el primer sistema de ficheros con journal en ser incluido en el núcleo estándar. También es el sistema de archivos por defecto en varias distribuciones, como Slackware, SuSE, Xandros, Yoper, Linspire, Kurumin Linux, FTOSX y Libranet.

ReiserFS ofrece funcionalidades que pocas veces se han visto en otros sistemas de archivos:

\* Journaling Esta es la mejora a la que se ha dado más publicidad, ya que previene el riesgo de

corrupción del sistema de archivos.

\* Reparticionamiento con el sistema de ficheros montado y desmontado. Podemos aumentar el tamaño del sistema de ficheros mientras lo tenemos montado y desmontado (online y offline). Para disminuirlo, únicamente se permite estando offline (desmontado). Namesys nos proporciona las herramientas para estas operaciones, e incluso, podemos usarlas bajo un gestor de volúmenes lógicos como LVM o EVMS.

- Tail packing, un esquema para reducir la fragmentación interna.

Algunas desventajas

\* Los usuarios que usen como sistema de ficheros EXT2, deben formatear sus discos, aunque no así los que usen EXT3.

\* ReiserFS v3 puede llegar a corromper el sistema de archivos cuando el árbol es reconstruido al realizar un chequeo. La reconstrucción del árbol es necesaria únicamente si el sistema de archivos se encuentra seriamente dañado (por ejemplo cuando se encuentra redimensionando y a causa de un corte de suministro eléctrico, el ordenador se apaga), lo que es algo nada habitual. Además, tampoco es recomendable almacenar imágenes en formato ReiserFS v3 (p.e. copias de seguridad, imágenes de disco para emuladores) en una partición también ReiserFS v3 sin haberlas antes convertido a un sistema que no confunda al programa encargado de realizar el chequeo de disco. Esto puede realizarse mediante la compresión o la encriptación de la imagen. Reformatear una partición existente de ReiserFS v3 puede dejar datos que también confundan al programa que realiza el chequeo, resultando en, por ejemplo, archivos del sistema anterior que aparecen en la reconstrucción actual del árbol.

\* ReiserFS en versiones del kernel anteriores a la 2.4.10 se considera inestable y no se recomienda su uso, especialmente en conjunción con NFS

\* Algunas operaciones sobre archivos (por ejemplo `unlink(2)`) no son síncronas bajo ReiserFS, lo que pueden causar comportamientos extraños en aplicaciones fuertemente basadas en locks de

archivos.

\* No se conoce una forma de desfragmentar un sistema de archivos ReiserFS, aparte de un volcado completo y su restauración.

- Tempranas implementaciones de ReiserFS (anteriores a la incluida en el kernel 2.6.2), eran susceptibles de problemas de escrituras fuera de orden, lo que provocaba que archivos siendo escritos durante una caída del sistema, ganaran un pico de bytes extras de basura en el siguiente montado del sistema de archivos. La implementación actual de journaling, es correcta en este aspecto, manteniendo el journaling ordenado, del estilo de EXT3.
- Para la asignación de espacio, ReiserFS usa bloques de tamaño fijo (4KB) que afecta negativamente al rendimiento en operaciones sobre ficheros grandes. El otro punto débil de ReiserFS es que el rendimiento sobre ficheros esparcidos (sparse, ficheros que no tienen todos los bloques de datos ocupados) es significativamente peor que Ext2, aunque Namesys está trabajando en este tema.

Más información:

<http://www.namesys.com/>

## Reiser4

Reiser4 es un sistema de archivos para computadoras. Se trata de la versión más reciente del sistema de archivos ReiserFS, implementada desde cero. Entre sus ventajas se encuentran:

- \* Soporte eficiente de gran cantidad de archivos pequeños.
- \* Manejo de directorios con cientos de miles de archivos.
- \* Infraestructura flexible que permite extensiones.
- \* Transacciones atómicas en la modificación del sistema de archivos.
- \* Manejo eficiente del diario por la vía de logs.
- \* Estructura del archivos dinámicamente optimizada.
- Integración de metadatos en el espacio de

nombres del sistema de archivos.

Más información:

<http://www.namesys.com/v4/v4.html>

## XFS

XFS es un sistema de archivos con journaling de alto rendimiento creado por SGI (antiguamente Silicon Graphics Inc.) para su implementación de UNIX llamada IRIX. En mayo del 2000, SGI liberó XFS bajo una licencia de código abierto.

Sus características más destacables son:

- \* Journaling muy cuidado y optimizado.
- \* Implementación paralelizada, que escala con el número de CPU's.
- \* Direccionamiento de 64 bits.
- \* Rendimiento y fiabilidad demostrada tras años de explotación comercial.

Todo esto hace de XFS un sistema de archivos altamente escalable y fiable.

Viene incorporado en las ramas 2.5.xx y 2.6.xx del kernel Linux, y solo estuvo disponible para la rama 2.4.xx como parche hasta que en la versión 2.4.25 Marcelo Tossati (responsable de la rama 2.4) lo consideró suficientemente estable para incorporarlo en la rama principal de desarrollo.

Existen también proyectos para incorporar XFS en FreeBSD.

Más información:<http://oss.sgi.com/projects/xfs/>

## JFS

JFS (journaling filesystem en inglés) es un sistema de archivos con respaldo de transacciones desarrollado por IBM y usado en sus servidores. Fue diseñado con la idea de conseguir "servidores de alto rendimiento y servidores de archivos de altas prestaciones, asociados a e-business". Según se lee en la documentación y el código fuente, va a pasar un tiempo antes de que la adaptación a Linux

este finalizada e incluida en la distribución estándar del kernel. JFS utiliza un método interesante para organizar los bloques vacíos, estructurándolos en un árbol y usa una técnica especial para agrupar bloques lógicos vacíos.

JFS fue desarrollado para AIX. La primera versión para Linux fue distribuida en el verano de 2000. La versión 1.0.0 salió a la luz en el año 2001. JFS está diseñado para cumplir las exigencias del entorno de un servidor de alto rendimiento en el que sólo cuenta el funcionamiento. Al ser un sistema de ficheros de 64 bits, JFS soporta ficheros grandes y particiones LFS (del inglés Large File Support), lo cual es una ventaja más para los entornos de servidor.

Las principales ventajas de JFS son:

- \* Eficiente respaldo de transacciones (Journaling).

JFS, al igual que ReiserFS, sigue el principio de metadata only. En vez de una completa comprobación sólo se tienen en cuenta las modificaciones en los metadatos provocadas por las actividades del sistema. Esto ahorra una gran cantidad de tiempo en la fase de recuperación del

sistema tras una caída. Las actividades simultáneas que requieren más entradas de protocolo se pueden unir en un grupo, en el que la pérdida de rendimiento del sistema de ficheros se reduce en gran medida mediante múltiples procesos de escritura.

- \* Eficiente administración de directorios.

JFS abarca diversas estructuras de directorios. En pequeños directorios se permite el almacenamiento directo del contenido del directorio en Inode. En directorios más grandes se utiliza Btrees, que facilitan considerablemente la administración del directorio.

- \* Mejor utilización de la memoria mediante adjudicación dinámica de Inodes.

Con Ext2 debe dar por anticipado el grosor del Inode (la memoria ocupada por la información de administración). Con ello se limita la cantidad máxima de ficheros o directorios de su sistema de ficheros. JFS le ahorra esto, puesto que asigna memoria Inode de forma dinámica y la pone a su disposición cuando no se está utilizando.

Más información: <http://www.ibm.com/>

A continuación te dejo algunas tablas jeje.

<b>Sistema de Archivo</b>	<b>Creador</b>	<b>Fecha</b>	<b><u>Sistema Operativo</u></b>
<b><u>DECtape</u></b>	<u>DEC</u>	<u>1964</u>	<u>PDP-6 Monitor</u>
<b><u>Level-D</u></b>	<u>DEC</u>	<u>1968</u>	<u>TOPS-10</u>
<b><u>V6FS</u></b>	<u>Bell Labs</u>	<u>1972</u>	<u>Version 6 Unix</u>
<b><u>RT-11</u></b>	<u>DEC</u>	<u>1973</u>	<u>RT-11</u>
<b><u>FAT12</u></b>	<u>Microsoft</u>	<u>1977</u>	<u>Microsoft Disk BASIC</u>
<b><u>V7FS</u></b>	<u>Bell Labs</u>	<u>1979</u>	<u>Version 7 Unix</u>
<b><u>ODS-2</u></b>	<u>DEC</u>	<u>1979</u>	<u>OpenVMS</u>
<b><u>FFS</u></b>	<u>Kirk McKusick</u>	<u>1983</u>	<u>4.2BSD</u>
<b><u>MFS</u></b>	<u>Apple Computer</u>	<u>1984</u>	<u>Mac OS</u>

<u><b>HFS</b></u>	<u>Apple Computer</u>	<u>1985</u>	<u>Mac OS</u>
<u><b>OFS54</b></u>	<u>Metacomco for Commodore</u>	<u>1985</u>	<u>Amiga OS</u>
<u><b>NWFS</b></u>	<u>Novell</u>	<u>1985</u>	<u>NetWare 286</u>
<u><b>Amiga FFS</b></u>	<u>Commodore</u>	<u>1987</u>	<u>Amiga OS 1.3</u>
<u><b>FAT16</b></u>	<u>Microsoft</u>	<u>1987</u>	<u>MS-DOS 3.31</u>
<u><b>HPFS</b></u>	<u>IBM &amp; Microsoft</u>	<u>1988</u>	<u>OS/2</u>
<u><b>JFS</b></u>	<u>IBM</u>	<u>1990</u>	<u>AIX11</u>
<u><b>VxFS</b></u>	<u>VERITAS</u>	<u>1991</u>	<u>SVR4.0</u>
<u><b>AdvFS</b></u>	<u>DEC</u>	<u>Before 1993</u>	<u>Digital Unix</u>
<u><b>NTFS</b></u>	<u>Microsoft, <a href="#">Gary Kimura</a>, <a href="#">Tom Miller</a></u>	<u>1993</u>	<u>Windows NT</u>
<u><b>LFS</b></u>	<u>Margo Seltzer</u>	<u>1993</u>	<u>Berkeley Sprite</u>
<u><b>ext2</b></u>	<u>Rémy Card</u>	<u>1993</u>	<u>Linux</u>
<u><b>UFS1</b></u>	<u>Kirk McKusick</u>	<u>1994</u>	<u>4.4BSD</u>
<u><b>XFS</b></u>	<u>SGI</u>	<u>1994</u>	<u>IRIX</u>
<u><b>UDF</b></u>	<u>ISO/ECMA/OSTA</u>	<u>1995</u>	-
<u><b>FAT32</b></u>	<u>Microsoft</u>	<u>1996</u>	<u>Windows 95b10</u>
<u><b>QFS</b></u>	<u>Sun Microsystems</u>	<u>1996</u>	<u>Solaris</u>
<u><b>Be File System</b></u>	<u>Be Inc., <a href="#">D. Giampaolo</a>, <a href="#">C. Meurillon</a></u>	<u>1996</u>	<u>BeOS</u>
<u><b>HFS Plus</b></u>	<u>Apple</u>	<u>1998</u>	<u>Mac OS 8.1</u>
<u><b>NSS</b></u>	<u>Novell</u>	<u>1998</u>	<u>NetWare 5</u>
<u><b>ext3</b></u>	<u>Stephen Tweedie</u>	<u>1999</u>	<u>Linux</u>
<u><b>JFS2</b></u>	<u>IBM</u>	<u>1999</u>	<u>OS/2 WSeB</u>
<u><b>GFS</b></u>	<u>Sistina(<a href="#">Red Hat</a>)</u>	<u>2000</u>	<u>Linux</u>
<u><b>ReiserFS</b></u>	<u>Namesys</u>	<u>2001</u>	<u>Linux</u>
<u><b>FATX</b></u>	<u>Microsoft</u>	<u>2002</u>	<u>Xbox</u>
<u><b>UFS2</b></u>	<u>Kirk McKusick</u>	<u>2002</u>	<u>FreeBSD 5.0</u>
<u><b>OCFS</b></u>	<u>Oracle</u>	<u>2002</u>	<u>Linux</u>
<u><b>ODS-5</b></u>	<u>DEC</u>	<u>2003</u>	<u>OpenVMS 8.0</u>
<u><b>Fossil</b></u>	<u>Bell Labs</u>	<u>2003</u>	<u>Plan 9 from Bell Labs 4</u>
<u><b>Google File System</b></u>	<u>Google</u>	<u>2003</u>	<u>Linux</u>
<u><b>ZFS</b></u>	<u>Sun Microsystems</u>	<u>2004</u>	<u>Solaris</u>
<u><b>Reiser4</b></u>	<u>Namesys</u>	<u>2004</u>	<u>Linux</u>
<u><b>OCFS2</b></u>	<u>Oracle</u>	<u>2005</u>	<u>Linux</u>

<a href="#">NILFS</a>	<a href="#">NTT</a>	<a href="#">2005</a>	<a href="#">Linux</a>
<a href="#">GFS2</a>	<a href="#">Red Hat</a>	<a href="#">2006</a>	<a href="#">Linux</a>

## Limites

Nombre	Tamaño maximo de nombre de archivo	Tamaño maximo del archivo	Tamaño maximo de partición
<a href="#">DECtape</a>	6.3	369,280 bytes (577 * 640)	369,920 Bytes (578 * 640)
<a href="#">Level-D</a>	6.3	34,359,738,368 words (2**35-1); 206,158,430,208 SIXBIT bytes	Approx 12 GB (64 * 178MB)
<a href="#">RT-11</a>	12 bytes	33,554,432 bytes (65536 * 512)	33,554,432 Bytes
<a href="#">V6FS</a>	14 bytes <a href="#">24</a>	<a href="#">8MiB</a> <a href="#">57</a>	<a href="#">2TiB</a>
<a href="#">V7FS</a>	14 bytes <a href="#">24</a>	<a href="#">1GiB</a> <a href="#">58</a>	<a href="#">2TiB</a>
<a href="#">FAT12</a>	255 bytes <a href="#">24</a>	<a href="#">32MiB</a>	<a href="#">1MiB</a> to <a href="#">32MiB</a>
<a href="#">FAT16</a>	255 bytes <a href="#">24</a>	<a href="#">2GiB</a>	<a href="#">16MiB</a> to <a href="#">2GiB</a>
<a href="#">FATX</a>	42 bytes <a href="#">24</a>	<a href="#">2GiB</a>	<a href="#">16MiB</a> to <a href="#">2GiB</a>
<a href="#">Fossil</a>	???	???	???
<a href="#">MFS</a>	255 bytes	<a href="#">256MiB</a>	<a href="#">256MiB</a>
<a href="#">HFS</a>	31 bytes	<a href="#">2GiB</a>	<a href="#">2TiB</a>
<a href="#">FAT32</a>	255 bytes <a href="#">24</a>	<a href="#">4GiB</a>	<a href="#">512MiB</a> to <a href="#">2TiB</a> <a href="#">7</a>
<a href="#">HPFS</a>	255 bytes	<a href="#">4GiB</a>	<a href="#">2TiB</a> <a href="#">13</a>
<a href="#">NTFS</a>	255 characters	<a href="#">16EiB</a> <a href="#">55</a>	<a href="#">16EiB</a> <a href="#">55</a>
<a href="#">HFS Plus</a>	255 UTF-16 characters <a href="#">1</a>	<a href="#">8EiB</a>	<a href="#">8EiB</a> <a href="#">71</a>
<a href="#">FFS</a>	255 bytes	<a href="#">4GiB</a>	<a href="#">256TiB</a>
<a href="#">UFS1</a>	255 bytes	<a href="#">4GiB</a> to <a href="#">256TiB</a>	<a href="#">256TiB</a>
<a href="#">UFS2</a>	255 bytes	<a href="#">512GiB</a> to <a href="#">32PiB</a>	<a href="#">1YiB</a>

<a href="#"><u>ext2</u></a>	255 bytes	16 <a href="#"><u>GiB</u></a> to 2 <a href="#"><u>TiB4</u></a>	2 <a href="#"><u>TiB</u></a> to 32 <a href="#"><u>TiB</u></a>
<a href="#"><u>ext3</u></a>	255 bytes	16 <a href="#"><u>GiB</u></a> to 2 <a href="#"><u>TiB4</u></a>	2 <a href="#"><u>TiB</u></a> to 32 <a href="#"><u>TiB</u></a>
<a href="#"><u>GFS</u></a>	255	2 <a href="#"><u>TB</u></a> to 8 <a href="#"><u>EB63</u></a>	2 <a href="#"><u>TB</u></a> to 8 <a href="#"><u>EB63</u></a>
<a href="#"><u>ReiserFS</u></a>	4032 bytes/255 characters	8 <a href="#"><u>TiB8</u></a>	16 <a href="#"><u>TiB</u></a>
<a href="#"><u>Reiser4</u></a>	???	8 <a href="#"><u>TiB</u></a> on x86	???
<a href="#"><u>OCFS</u></a>	255 bytes	8 <a href="#"><u>TiB</u></a>	8 <a href="#"><u>TiB</u></a>
<a href="#"><u>OCFS2</u></a>	255 bytes	4 <a href="#"><u>PiB</u></a>	4 <a href="#"><u>PiB</u></a>
<a href="#"><u>XFS</u></a>	255 bytes	8 <a href="#"><u>EiB9</u></a>	8 <a href="#"><u>EiB9</u></a>
<a href="#"><u>JFS</u></a>	255 bytes	8 <a href="#"><u>EiB</u></a>	512 <a href="#"><u>TiB</u></a> to 4 <a href="#"><u>PiB</u></a>
<a href="#"><u>JFS2</u></a>	255 bytes	4 <a href="#"><u>PiB</u></a>	32 <a href="#"><u>PiB</u></a>
<a href="#"><u>QFS</u></a>	255 bytes	16 <a href="#"><u>EiB 72</u></a>	4 <a href="#"><u>PiB 72</u></a>
<a href="#"><u>Be File System</u></a>	255 bytes	12288 bytes to 260 <a href="#"><u>GiB3</u></a>	256 <a href="#"><u>PiB</u></a> to 2 <a href="#"><u>EiB</u></a>
<a href="#"><u>AdvFS</u></a>	255 characters	16 <a href="#"><u>TiB</u></a>	16 <a href="#"><u>TiB</u></a>
<a href="#"><u>NSS</u></a>	256 characters	8 <a href="#"><u>TiB</u></a>	8 <a href="#"><u>TiB</u></a>
<a href="#"><u>NWFS</u></a>	80 bytes <a href="#"><u>52</u></a>	4 <a href="#"><u>GiB</u></a>	1 <a href="#"><u>TiB</u></a>
<a href="#"><u>ODS-5</u></a>	236 bytes <a href="#"><u>15</u></a>	1 <a href="#"><u>TiB</u></a>	1 <a href="#"><u>TiB</u></a>
<a href="#"><u>VxFS</u></a>	255 bytes	16 <a href="#"><u>EiB</u></a>	???
<a href="#"><u>UDF</u></a>	255 bytes	16 <a href="#"><u>EiB</u></a>	???
<a href="#"><u>ZFS</u></a>	255 bytes	16 <a href="#"><u>EiB</u></a>	16 <a href="#"><u>EiB</u></a>

## Características

	<a href="#"><u>Enlaces Fuertes</u></a>	<a href="#"><u>Enlaces Suaves</u></a>	<a href="#"><u>Journal de bloque</u></a>	<a href="#"><u>Journal de Metadata</u></a>	<a href="#"><u>Sensible Mayusculas y Minusculas</u></a>	<a href="#"><u>Registro de cambio de archivo</u></a>
<a href="#"><u>DECtape</u></a>	No	No	No	No	No	No
<a href="#"><u>Level-D</u></a>	No	No	No	No	No	No

<u><b>RT-11</b></u>	No	No	No	No	No	No
<u><b>V6FS</b></u>	Yes	No	No	No	Yes	No
<u><b>V7FS</b></u>	Yes	No <u>59</u>	No	No	Yes	No
<u><b>FAT12</b></u>	No	No	No	No	No	No
<u><b>FAT16</b></u>	No	No	No	No	No	No
<u><b>FAT32</b></u>	No	No	No	No	No	No
<u><b>HPFS</b></u>	No	No	No	No	No	No
<u><b>NTFS</b></u>	Yes	Partial <u>34</u>	No	Yes	Yes <u>36</u>	Yes
<u><b>HFS Plus</b></u>	Partial	Yes	No	Yes <u>48</u>	Partial <u>35</u>	Yes <u>64</u>
<u><b>FFS</b></u>	Yes	Yes	No	No	Yes	No
<u><b>UFS1</b></u>	Yes	Yes	No	No	Yes	No
<u><b>UFS2</b></u>	Yes	Yes	No	No <u>66</u>	Yes	No
<u><b>LFS</b></u>	Yes	Yes	Yes <u>38</u>	No	Yes	No
<u><b>ext2</b></u>	Yes	Yes	No	No	Yes	No
<u><b>ext3</b></u>	Yes	Yes	Yes <u>62</u>	Yes	Yes	No
<u><b>ReiserFS</b></u>	Yes	Yes	Yes <u>44</u>	Yes	Yes	No
<u><b>Reiser4</b></u>	Yes	Yes	Yes	No	Yes	No
<u><b>OCFS</b></u>	No	Yes	No	No	Yes	No
<u><b>OCFS2</b></u>	Yes	Yes	Yes	Yes	Yes	No
<u><b>XFS</b></u>	Yes	Yes	No	Yes	Yes <u>40</u>	Yes
<u><b>JFS</b></u>	Yes	Yes	No	Yes	Yes <u>30</u>	No
<u><b>QFS</b></u>	Yes	Yes	No	No	Yes	No
<u><b>Be File System</b></u>	Yes	Yes	No	Yes	Yes	???
<u><b>NSS</b></u>	Yes	Yes	???	Yes	Yes <u>20</u>	Yes <u>6</u>
<u><b>NWFS</b></u>	Yes <u>53</u>	Yes <u>53</u>	No	No	Yes <u>20</u>	Yes <u>6</u>
<u><b>ODS-2</b></u>	Yes	Yes <u>18</u>	No	Yes	No	Yes
<u><b>ODS-5</b></u>	Yes	Yes <u>18</u>	No	Yes	No	Yes
<u><b>UDF</b></u>	Yes	Yes	Yes <u>38</u>	Yes <u>38</u>	Yes	No
<u><b>VxFS</b></u>	Yes	Yes	Yes	No	Yes	Yes
<u><b>Fossil</b></u>	No	No	No	No	Yes	Yes
<u><b>ZFS</b></u>	Yes	Yes	Yes <u>56</u>	No <u>56</u>	Yes	No

### En conclusión

Yo recomendaría el uso de XFS para ser instalado en cualquier máquina :

- \* Utiliza la máxima capacidad de los discos duros
- \* Es el más rápido en crear, montar y desmontar el sistema de archivos
- \* Es el sistema de archivos más veloz para operaciones en archivos grandes (mayores a 500 Mb)
- \* Es un gran segundo lugar para operaciones en una gran cantidad de archivos o directorios pequeños o de mediano tamaño
- \* Demuestra un buen equilibrio entre uso de CPU y tiempo de respuesta para grandes listados de directorios o búsquedas de archivos
  - No es el sistema de archivos con menos uso de CPU pero se observa un aceptable uso de los recursos del sistema para hardware pasado de moda y nuevo!
  - XFS está siendo usado en grandes servidores, especialmente en la industria del cine y los efectos especiales!

## **Links**

<http://www.wikipedia.com>

[http://www.wikilearning.com/rendimiento\\_y\\_conclusiones-wkccp-994-18.htm](http://www.wikilearning.com/rendimiento_y_conclusiones-wkccp-994-18.htm)

<http://bulma.net/body.phtml?nIdNoticia=626>

## **CONTACTO**

Armando Rodriguez Arguijo

<http://www.geocities.com/armando889>

armando889@yahoo.com