

SISTEMAS DE ARCHIVOS EXTENDED

ext, ext2, ext3, ext4

EXTENDED FILE SYSTEM (EXT)

EXTENDED FILE SYSTEM (EXT)

Implementado en abril de 1992.

El primero de una serie de sistemas de archivos creados exclusivamente para Linux.

Diseñado por Rémy Card para superar algunas limitaciones del MINIX FS.

El primero en utilizar el API del Sistema Virtual de Archivos.

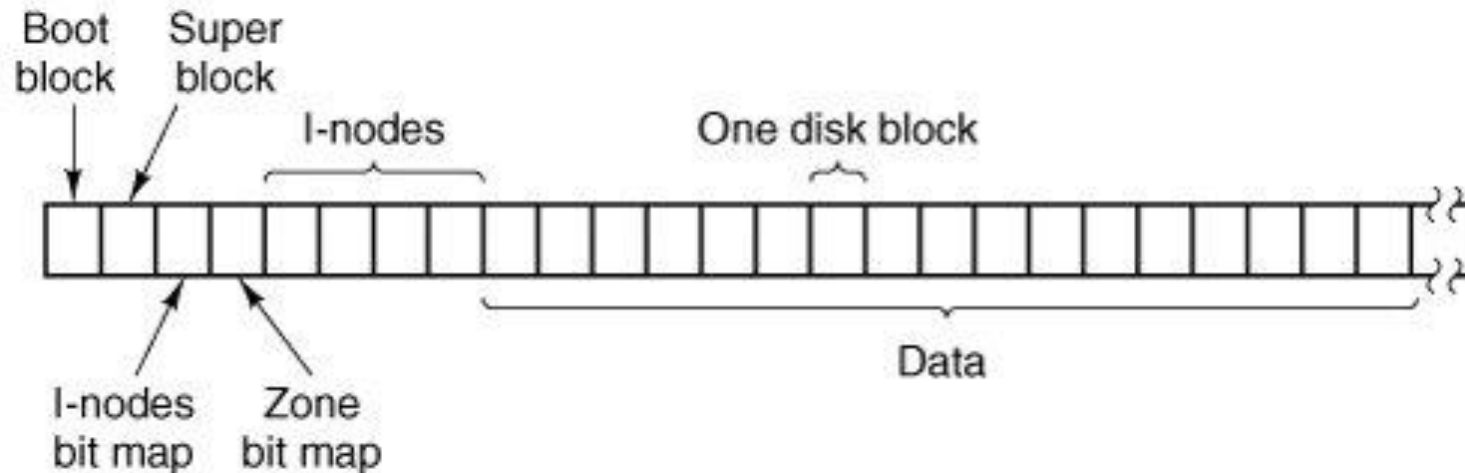


Fig.1 - Estructura del sistema de archivos MiniX.

SECOND EXTENDED FILE SYSTEM (EXT2)

BREVE HISTORIA

Desarrollado en enero de 1993 para el Kernel de Linux 0.99.

Survió como una solución a tres problemas que tenía ext:

- Modificación de inodo
- Modificación de datos
- Marcas de tiempo (timestamps) para acceso de archivo.

ESTRUCTURA EN DISCO

Hecho bajo la premisa que toda la información que exista en los archivos debe ser mantenida en bloques de datos.

Estos bloques son todos de la misma longitud y están agrupados en grupos de bloques.

Cada grupo contiene:

- Una copia del super bloque
- Su tabla de descriptor de grupo.
- Su bitmap de bloque
- Su bitmap de inodos
- Su tabla de inodos
- Los bloques de datos.

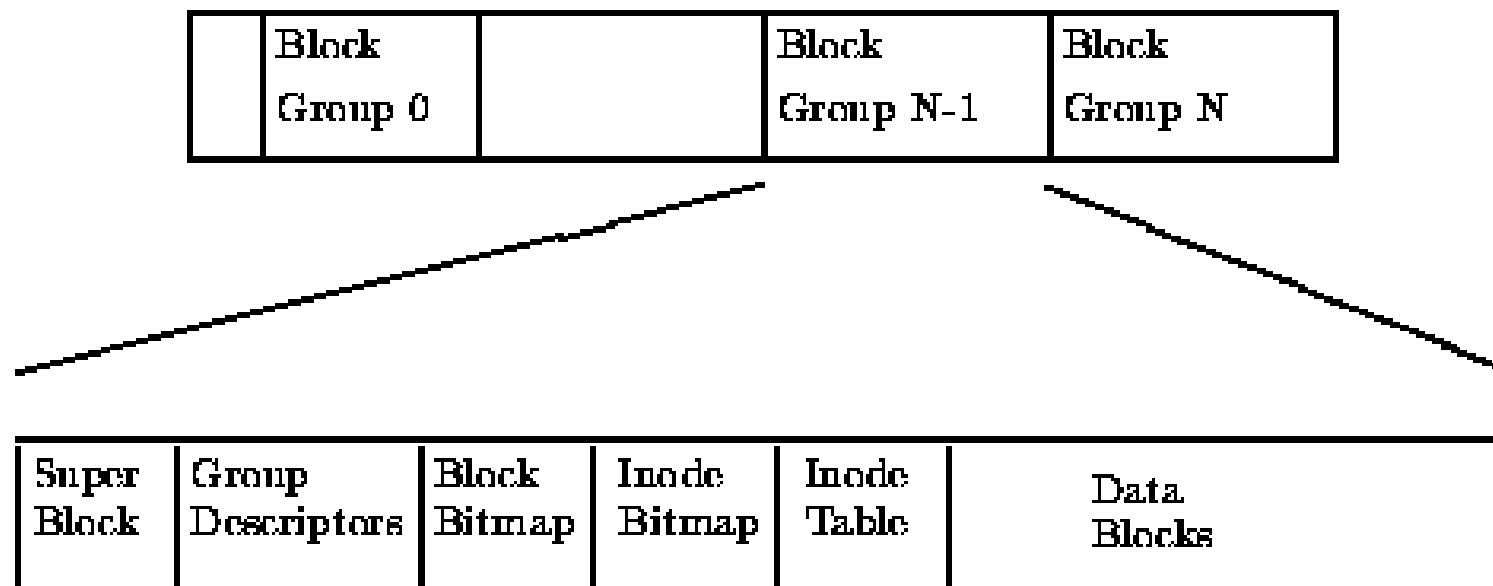


Fig.2 - Estructura de un bloque en ext2.

EL SUPERBLOQUE

Contiene una descripción del tamaño básico y la forma de este sistema de archivos.

Normalmente este archivo se lee del Grupo de Bloques 0 cuando el sistema es montado.

Cada Bloque de Grupo contiene una copia duplicada en caso de corrupción.

Contiene, entre otra información, lo siguiente:

- Número Mágico
- Numero de Bloques de Grupo
- Tamaño de bloque
- Bloques por Grupo
- Bloques libres
- Inodos libres
- Primer inodo

DESCRIPTOR DE GRUPO

Cada Grupo contiene una estructura de datos que lo describe.

Cada Descriptor de Grupo contiene, entre otra información, lo siguiente:

- Dirección de inicio del Bitmap del Bloque
- Dirección de inicio del Bitmap de Inodo
- Dirección de inicio de Tabla de Inodos

BITMAP DE BLOQUE Y BITMAP DE INODOS

Bitmap de bloque

Es una estructura que representa el estado actual de un Bloque dentro de un Grupo de Bloques. Cada bit representa dicho estado, un 1 significa “en uso” y un 0 significa “libre/disponible”.

Bitmap de inodos

El Bitmap de inodos funciona de manera similar al Bitmap de Bloque, con la diferencia de que cada bit representa un inodo en la Tabla de Inodos en lugar de un bloque.

INODOS Y SU TABLA

La tabla de inodos es usada para mantener registro de cada directorio, archivo regular, enlace simbólico o archivo especial.

Los inodos son el bloque de construcción básico; cada archivo y directorio en el sistema de archivos es descrito por un y sólo un inodo.

El inodo incluye información acerca del tamaño, permiso, dueño y localización en disco de un archivo o directorio.

Los inodos no guardan la información del nombre del archivo, esta información se guarda en los directorios.

INODOS

Contienen, entre otra información, lo siguiente:

- Modo. Describe el archivo y sus derechos de acceso.
- Tamaño en bytes del archivo.
- Marcas de tiempo.
- Número de bloques que ocupa.

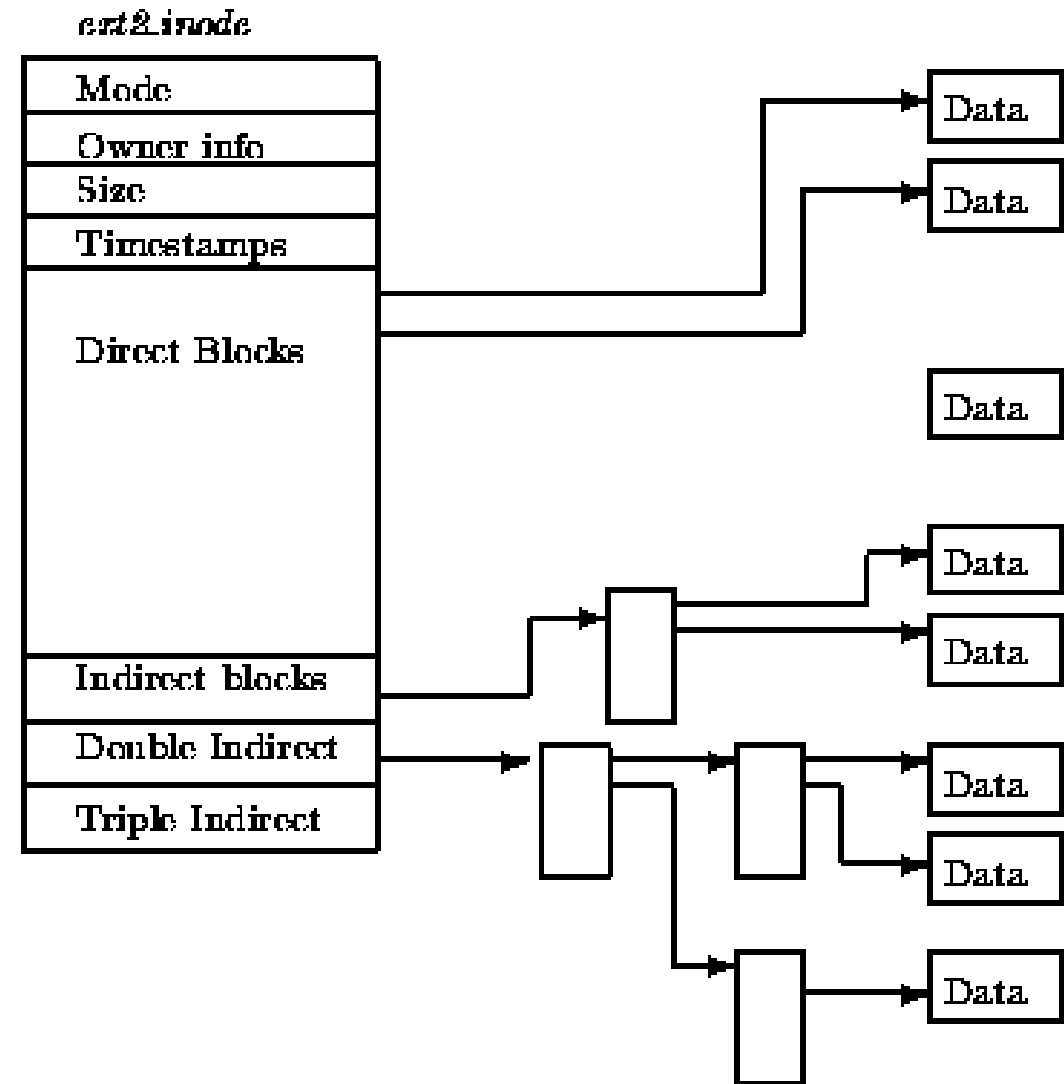


Fig. 3 - Estructura de un inodo en ext2.

DIRECTORIOS

Son archivos especiales que son usados para crear y guardar rutas de acceso a los archivos en el sistema de archivos.

Se utilizan para ordenar jerárquicamente archivos.

Cada directorio puede contener otros directorios, archivos regulares y archivos especiales.

Cada directorio contiene la siguiente información:

- El inodo de esta entrada de directorio.
- La longitud de esta entrada del directorio en bytes
- El nombre de esta entrada del directorio.

Las primeras dos entradas para cada directorio siempre son el estándar “.” y “..” (“este directorio” y el “directorio padre”, respectivamente).

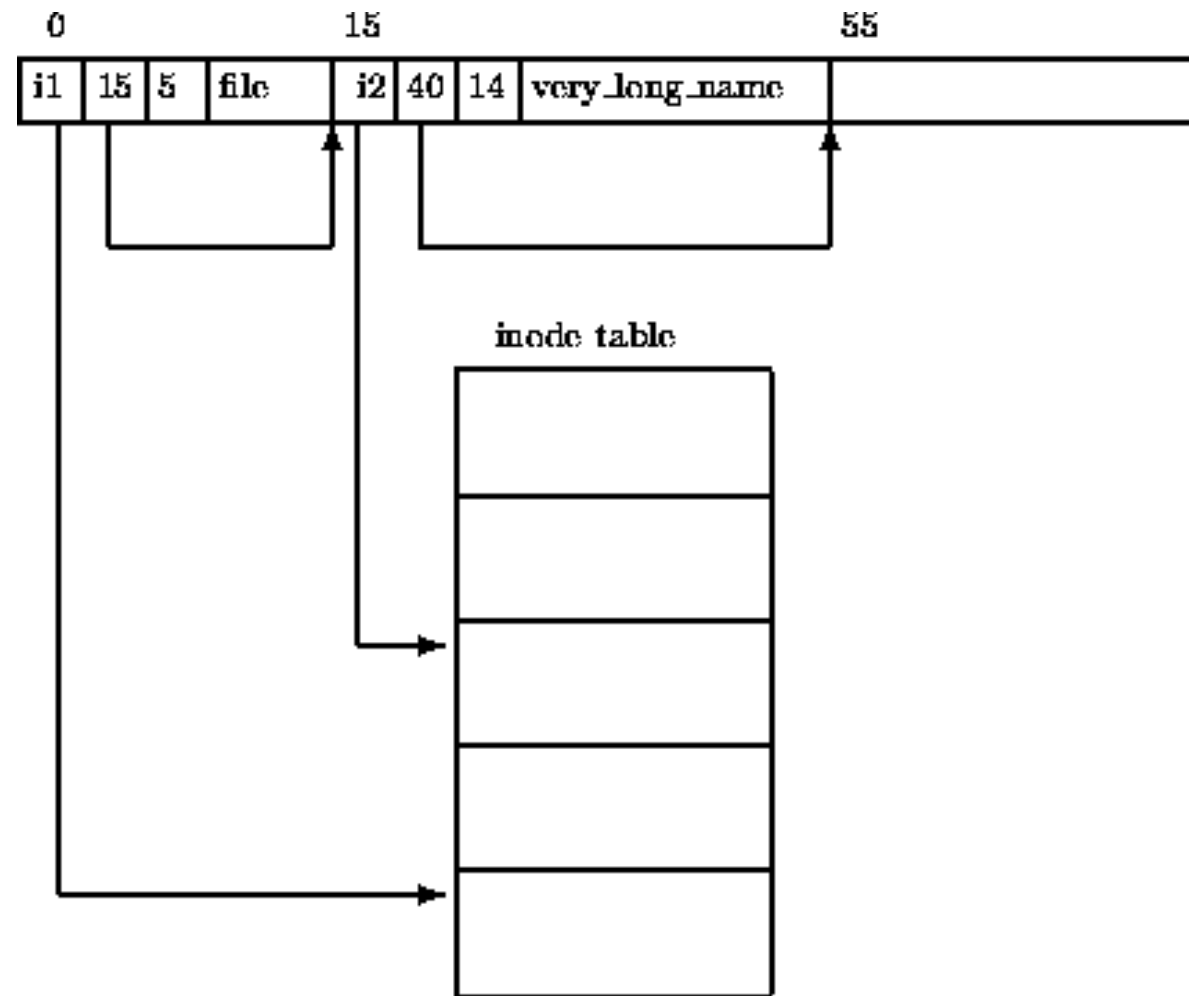


Fig 4. - Estructura de un directorio ext2

THIRD EXTENDED FILE SYSTEM (EXT3)

BREVE HISTORIA

Ext3 es un sistema de archivos con registro por diario (journaled file system).

Se unió al kernel principal de Linux en noviembre de 2001.

Desarrollado por Stephen Tweedie.

Tiene la ventaja de que los usuarios pueden actualizar desde ext2 sin tener que realizar copias de seguridad ni restaurar datos.

El sistema ext3, añade las siguientes características ext2.

- Un registro de diario (journal).
- Estructura de Arbol Hash para indexar grandes directorios.
- Crecimiento en línea del sistema de archivos.

JOURNALED FILE SYSTEM (JFS)

Son sistemas que mantienen registro de los cambios que aún no han sido entregados a la parte principal del sistema de archivos.

Consisten en un registro de diario en el que se almacena la información necesaria para restablecer los datos afectados por la transacción en caso de que esta falle.

Como los registros en el diario son escritos antes de que los cambios al sistema de archivos estén hechos, y como el sistema de archivos mantiene estos registros hasta que los cambios han sido aplicados al sistema de archivos, los sistemas con registro por diario maximizan la consistencia y minimizan el tiempo de reinicio en caso de un apagado de sistema inapropado.

LÍMITES EN TAMAÑO DE EXT3

El número máximo de bloques para ext3 es 2^{32} .

El tamaño del bloque puede variar, afectando máximo de archivos y al tamaño máximo del sistema de archivos.

Tamaño del bloque	Tamaño máximo de archivo	Tamaño máximo del sistema de archivos
1 KB	16 GB	2 TB
2 KB	256 GB	8 TB
4 KB	2 TB	16 TB

NIVELES DEL JOURNALING

Journal. Los metadatos y los ficheros de contenido son copiados al diario antes de ser llevados al sistema de archivos principal. Es el modo más lento y seguro.

Ordered. Este modo solo registra los cambios en los metadatos del sistema de archivos, pero vacía las actualizaciones de datos de archivos en el disco antes de realizar cambios en los metadatos del sistema de archivos asociado. Es el modo por defecto en ext3.

Writeback. Este modo solo registra los cambios en los metadatos del sistema de archivos. Este modo depende de que el proceso de escritura del sistema de archivos escriba los cambios de datos de archivos en el disco. Es el modo más rápido y riesgoso.

FOURTH EXTENDED FILE SYSTEM (EXT4)

BREVE HISTORIA

Este sistema se desarrolló como el sucesor de ext3.

Ext4 fue inicialmente una serie de extensiones retrocompatibles con ext3, muchos de ellos originalmente hechos por Cluuster File Systems para el sistema de archivos Lustre.

Algunos desarrolladores propusieron bifurcar ext3, renombrarlo ext4 y desarrollar todos los cambios ahí, esto con el fin de obtener mejor estabilidad.

La propuesta fue aceptada y, el 26 de junio de 2006, Theodore Ts'o anunció el nuevo plan de desarrollo para ext4.

El sistema de archivos ext4 fue lanzado el 25 de diciembre de 2008.

Este sistema de archivos principalmente mejora el rendimiento, la confiabilidad y la capacidad.

TAMAÑOS MÁXIMOS (32 BITS)

File System Maximums

Item	32-bit mode			
	1KiB	2KiB	4KiB	64KiB
Blocks	2 ³²	2 ³²	2 ³²	2 ³²
Inodes	2 ³²	2 ³²	2 ³²	2 ³²
File System Size	4TiB	8TiB	16TiB	256PiB
Blocks Per Block Group	8,192	16,384	32,768	524,288
Inodes Per Block Group	8,192	16,384	32,768	524,288
Block Group Size	8MiB	32MiB	128MiB	32GiB
Blocks Per File, Extents	2 ³²	2 ³²	2 ³²	2 ³²
Blocks Per File, Block Maps	16,843,020	134,480,396	1,074,791,436	4,398,314,962,956 (really 2 ³² due to field size limitations)
File Size, Extents	4TiB	8TiB	16TiB	256TiB
File Size, Block Maps	16GiB	256GiB	4TiB	256TiB

TAMAÑOS MÁXIMOS (64 BITS)

File System Maximums

Item	64-bit mode			
	1KiB	2KiB	4KiB	64KiB
Blocks	2 ⁶⁴	2 ⁶⁴	2 ⁶⁴	2 ⁶⁴
Inodes	2 ³²	2 ³²	2 ³²	2 ³²
File System Size	16ZiB	32ZiB	64ZiB	1YiB
Blocks Per Block Group	8,192	16,384	32,768	524,288
Inodes Per Block Group	8,192	16,384	32,768	524,288
Block Group Size	8MiB	32MiB	128MiB	32GiB
Blocks Per File, Extents	2 ³²	2 ³²	2 ³²	2 ³²
Blocks Per File, Block Maps	16,843,020	134,480,396	1,074,791,436	4,398,314,962,956 (really 2 ³² due to field size limitations)
File Size, Extents	4TiB	8TiB	16TiB	256TiB
File Size, Block Maps	16GiB	256GiB	4TiB	256TiB

CARACTERÍSTICAS

Extents

Los extents fueron introducidos para remplacar al esquema de bloques usado por los sistemas de archivos ext2 y ext3. Un extent es un conjunto de bloques físicos contiguos. Esto mejora el rendimiento al trabajar con ficheros de gran tamaño y reduce la fragmentación.

Asignación persistente de espacio en el disco

Esta función permite prealocar memoria en el disco, esto garantiza que el espacio reservado para un fichero estará disponible y, con mucha probabilidad, será contiguo.

Asignación retrasada de espacio en el disco

Ext4 hace uso de una técnica de mejora de rendimiento llamada Allocate-on-flush, también conocida como reserva de memoria retrasada.

Consiste en retrasar la reserva de bloques de memoria hasta que la información esté a punto de ser escrita en el disco, a diferencia de otros sistemas de archivos, los cuales reservan los bloques necesarios antes de ese paso.

Esto mejora el rendimiento y reduce la fragmentación al mejorar las decisiones de reserva de memoria basada en el tamaño real del fichero, pero aumenta el riesgo de pérdida de datos en sistemas antiguos.

Checksum en el journal

Tener checksums en el journal permite al sistema de archivos darse cuenta de que algunas de sus entradas no son válidas o están fuera de orden en el primer montaje después de un crasheo.

Esto evita el error de deshacer entradas de diario parciales o fuera de orden y dañar aún más el sistema de archivos.

Desfragmentación en línea

Esta característica permite desfragmentar la unidad mientras esta esté montada.

Comprobación del sistema de ficheros más rápido

En ext4, los grupos de bloques no asignados y secciones de la tabla de inodos están marcados como tales.

Esto permite a e2fsck (la herramienta para comprobar errores en ext2) saltárselos completamente en las comprobaciones y en gran medida reduce el tiempo requerido para comprobar un sistema de archivos del tamaño para el que ext4 está preparado.

Asignador multibloque

Ext4 asigna múltiples bloques para un fichero en una sola operación, lo cual reduce la fragmentación al intentar elegir bloques contiguos en el disco.

Timestamps mejorados

Ext3 ofrece marcas de tiempo granulares en segundos.

Ext4 ofrece marcas de tiempo en el orden de los nanosegundos.

Los sistemas de archivos Ext3 no proporcionaron suficientes bits para almacenar fechas posteriores al 18 de enero de 2038.

Ext4 agrega dos bits adicionales aquí, extendiendo la época de Unix otros 408 años.

CONCLUSIONES

GRACIAS POR SU ATENCIÓN