**EXT4**

**Организация файловой системы в Linux**

Корневая файловая система Linux имеет единую древовидную структуру

Так в Windows вы могли разбить жесткий диск компьютера на два логических диска С и D. У каждого логического диска был свой корневой каталог (папка), который обозначался C:\ и D:\ соответственно.

Файлами является всё:

* Жесткие диски и разделы созданные на них
* Съемные носители информации
* Папки

Так как в linux отсутствует буквенное обозначение дисков, все носители информации (ЖД, флешки), должны быть подключены в файловую систему linux в виде файлов, для того чтобы с ними можно было работать.

Процесс подключения носителя информации к файловой системе называется монтированием (mount). В Linux он автоматизирован и не требует вашего вмешательства.

**Структура файловой системы**

Как и в любой файловой системе UNIX, в составе ext2 можно выделить следующие составляющие:

* блоки и группы блоков;
* индексный дескриптор;
* суперблок.

Всё пространство раздела диска разбивается на блоки фиксированного размера, кратные размеру сектора.

Суперблок — основной элемент файловой системы. Он содержит общую информацию о файловой системе: общее число блоков и индексных дескрипторов в файловой системе, число свободных блоков и индексных дескрипторов в файловой системе, размер блока файловой системы, количество блоков и индексных дескрипторов в группе блоков, размер индексного дескриптора, идентификатор файловой системы.

Суперблок находится в 1024 байтах от начала раздела. От целостности суперблока напрямую зависит работоспособность файловой системы. Операционная система создаёт несколько резервных копий суперблока на случай повреждения раздела.

Сразу же после суперблока находится глобальная таблица дескрипторов группы блоков (Group Description Table). В ней описаны первый и последний блоки для каждой группы блоков, а также информация где именно в каждой группе начинается таблица inodes, начало блоков данных и т.д.

Битовая карта блока (**Block Bitmap**) – это специальная таблица, в которой указано какие блоки в группе использованы, а какие свободны. Эта информация используется во время распределения информации в блоке. 0 – означает что блок свободен, а 1 – что занят.

Битовая карта inodes (**Inode Bitmap**) – эта таблица аналогична битовой карте блока, только в ней отображается информация о свободных inodes, которые могут быть использованы для записи новых файлов.

Таблица индексных дескрипторов

Блоки данных – выделенные физические блоки памяти, в которых хранятся данные пользователя.

Базовым понятием файловой системы является индексный дескриптор, или inode.

Они хранят всю информацию, связанную с файлом, кроме имени файла и фактических данных.

Атрибутами файла являются его тип (обычный файл, каталог и т.д.), права доступа к нему, идентификаторы владельца и группы, размер, время создания.

Информация о физическом расположении зависит от размеров файла и представляет собой последовательность абсолютных номеров блоков, содержащих данные файла. Эта информация представляет собой последовательность 32-битных номеров блоков, содержащих данные файла.

Первые 12 номеров – это прямые ссылки на информационные блоки (direct blocks number). 13-й номер является косвенной ссылкой (indirect blocks number). В нём находится адрес блока, в котором хранятся адреса информационных блоков.14-й номер – двойная косвенная ссылка (double indirect blocks number), 15-й номер – тройная косвенная ссылка (triple indirect blocks number)

inode содержит информацию о «физическом» расположении данных. Связь логического имени файла с inode производится с помощью жёстких ссылок.

**Файловые системы в Linux**

В отличие от Windows, в Linux широкий выбор ФС

В Linux несколько предустановленных и доступных файловых систем. В зависимости от целей и задач пользователя (а также достоинств и недостатков самих ФС) можно выбрать любую файловую систему, доступную в дистрибутиве ОС.

Другие файловые системы, такие как FAT, NTFS могут использоваться в Linux, но не в качестве корневой файловой системы, потому что они для этого не предназначены.

**Ext4 общие сведения**

Extended File System – расширенная файловая система. Первая файловая система разработанная специально для ОС на ядре Linux. Целью было преодолеть ограничения файловой системы Minix File System, связанные в первую очередь с размером файла.

Файловая система Ext4 была принята в качестве файловой системы по умолчанию более десяти лет и зарекомендовала себя как надежная и стабильная файловая система.

* 260 Байт – максимальный размер раздела
* 244 Байт – максимальный размер файла

**Особенности Ext4**

* Обратная совместимость:

Обратная совместимость — преимуществом файловой системы Ext4 является ее обратная совместимость с системами Ext3 и Ext2. Это повышает производительность и гибкость FS. Так как некоторые функции Ext4 могут быть реализованы в файловых системах Ext3 и Ext2. И так же файловые системы Ext3 и Ext2 могут быть смонтированы как Ext4.

* Неограниченное количество подкаталогов
* Улучшенные временные метки:

в Ext4 реализованы временные метки, измеряемые в наносекундах, что является улучшением по сравнению с детализацией временных меток на основе секунд.

* Журналируемая файловая система

Журналирование – процесс, при котором файловая система ведет постоянный учет всех происходящих на диске изменений

Благодаря данному механизму после сбоя файловая система автоматически возвращается в рабочее состояние.

Журналирование обеспечивает целостность файловой системы, однако оно приводит к дополнительным накладным расходам на дисковые операции. Потому если в особых ситуация журналирования является излишней функций, файловая система Ext4 позволяет его отключить, что приводит к небольшому приросту производительности.

* Экстенты

Дисковые ресурсы для файлов выделялись с помощью битовых карт свободного места - способа, не выделяющегося ни скоростью, ни масштабируемостью. Поэтому для улучшения выделения ресурсов и поддержки более эффективной структуры хранения данных в ext4 вместо битовых карт применяются экстенты. Экстент - это просто способ представления непрерывной последовательности блоков памяти. При использовании экстентов сокращается количество метаданных, так как вместо информации о том, где находится каждый блок памяти, экстенты содержат информацию о том, где находится большой список непрерывных блоков памяти.

**Отложенное выделение блоков памяти**

**Выделение блоков памяти группами**

**Дефрагментация "на лету"**