

4.15 Файловые системы. Древовидные и сетевые ФС. Журналирующие ФС. Реализация ФС. Файловые системы FAT, NTFS, Ext.

Файловая система (file system) — регламент, определяющий способ организации, хранения и именования данных на **носителях информации**. Она определяет **формат** физического хранения файлов, определяет размер имени файла, максимальный возможный размер файла, набор атрибутов файла. Некоторые файловые системы предоставляют сервисные возможности, например, **разграничение доступа** или **шифрование** файлов.

Файловая система связывает **носитель информации** с одной стороны и **API** для доступа к файлам — с другой. Когда прикладная программа обращается к файлу, она не имеет никакого представления о том, в каких секторах диска расположен файл, на каком типе диска (**CD**, **жестком диске**) записан. Все, что знает программа — это имя файла, его размер и атрибуты. Эти данные она получает от **драйвера** файловой системы. Именно файловая система устанавливает, где и как будет записан файл на диске.

С точки зрения ОС весь диск представляет из себя набор кластеров размером от 512 байт и выше. Драйверы файловой системы организуют кластера в **файлы** и каталоги (реально являющиеся файлами содержащими список файлов в этом каталоге). Эти же драйвера отслеживают какие из кластеров в настоящее время используются, какие свободны, какие помечены как неисправные.

Однако файловая система не обязательно связана с физическим **носителем информации** (диском). Существуют виртуальные и сетевые файловые системы, которые являются лишь способом доступа к файлам, находящимся на удаленном компьютере.

Классификация файловых систем

По предназначению файловые системы можно классифицировать на следующие категории:

- для носителей с **произвольным доступом** (например, **жесткий диск**): **FAT32**, **HPFS**, **EXT2** и др. В последнее время широкое распространение получили **журналируемые файловые системы** такие как **EXT3**, **ReiserFS**, **JFS**, **NTFS**, **XFS** и др.
- для носителей с **последовательным доступом** (например, **магнитные ленты**): **QIC** и др.
- для оптических носителей — **CD** и **DVD**: **ISO9660**, **ISO9690**, **HFS**, **UDF** и др.
- виртуальные файловые системы: **AEFS** и др.
- Сетевые файловые системы: **NFS**, **SMBFS** и др.

Журналируемая файловая система

Материал из Википедии — свободной энциклопедии

Перейти к: [навигация](#), [поиск](#)

Журналируемые файловые системы — это класс **файловых систем**, характерная черта которых — ведение журнала, хранящего список изменений, в той или иной степени помогающего сохранить целостность файловой системы.

Запуск проверки системы (например, **fsck**) на больших файловых системах может занять много времени, что

очень плохо для сегодняшних высокоскоростных систем. Причиной отсутствия целостности в файловой системе может быть некорректное **размонтирование**, например, если в момент прекращения работы на диск велась запись. Приложения могли обновлять данные, содержащиеся в файлах, и система могла обновлять **метаданные** файловой системы, которые являются «данными о данных файловой системы», иными словами, информация о том, какие блоки связаны с какими файлами, какие файлы размещены в каких директориях и тому подобное. Ошибки (отсутствие целостности) в файлах данных — это плохо, но куда хуже ошибки в метаданных файловой системы, что может привести к потерям файлов и другим серьезным проблемам.

Для минимизации проблем, связанных с целостностью, и минимизации времени перезапуска системы, журналируемая файловая система хранит список изменений, которые она будет проводить с файловой системой перед фактической записью изменений. Эти записи хранятся в отдельной части файловой системы, называемой «журналом», или «логом». Как только изменения файловой системы безопасно внесены в журнал, журналируемая файловая система применяет эти изменения к файлам или метаданным, а затем удаляет эти записи из журнала. Записи журнала организованы в наборы связанных изменений файловой системы, что очень похоже на то, как изменения добавляемые в **базу данных** организованы в **транзакции**.

Наличие журнала повышает вероятность сохранения целостности файловой системы, потому что записи в **лог-файл** ведутся до проведения фактических изменений, и эти записи хранятся до тех пор, пока они не будут целиком и безопасно применены. При перезагрузке компьютера программа монтирования может гарантировать целостность журналируемой файловой системы простой проверкой лог-файла на наличие ожидаемых, но не произведенных изменений и последующей записью их в файловую систему. Т.о. при наличии журнала в большинстве случаев системе не нужно проводить проверку целостности файловой системы, а это означает, что компьютер будет доступен для работы практически сразу после перезагрузки. Соответственно, шансы потери данных в связи с проблемами в файловой системе значительно снижаются.

Существует несколько журналируемых файловых систем, доступных в **Linux**. Наиболее известные из них:

- **XFS**, журналируемая файловая система разработанная **Silicon Graphics**, но сейчас выпущенная открытым кодом (open source);
- **ReiserFS**, журналируемая файловая система разработанная специально для Linux;
- **JFS**, журналируемая файловая система первоначально разработанная **IBM**, но сейчас выпущенная как открытый код;
- **ext3** — журналируемое расширение файловой системы **ext2**, используемой на большинстве версий **GNU/Linux**. Уникальная особенность системы ext3 — возможность перехода на неё с ext2 без переформатирования диска. Разработана доктором Стефаном Твиди (Stephan Tweedie).

В семействе ОС **Microsoft Windows** к журналируемым относится файловая система **NTFS**. В **Mac OS X** — **HFS+**.

FAT

	FAT12	FAT16	FAT32
Разработчик		Microsoft	
Полное название	File Allocation Table (12-bit версия)	File Allocation Table (русск. Таблица Размещения Файлов) (16-bit версия)	
Представлена	1977 (Microsoft Disk BASIC)	Июль 1988 (MS-DOS 4.0)	Август 1996 (Windows 95 OSR2)
Идентификатор тома	0x01 (MBR)	0x04, 0x06, 0x0E (MBR)	0x0B, 0x0C (MBR) EBD0A0A2-B9E5-4433-87C0-68B6B72699C7 (GPT)
Структуры			
Содержимое директории	Таблица		

Размещение файлов Сбойные блоки		Связанный список Тегирование кластера	
Ограничения			
Максимальный размер файла	32 MiB	2 GiB	4 GiB
Максимальное количество файлов	4.077	65.517	268.435.437
Максимальная длина имени файла	8.3, или 255 символов при использовании LFN		
Максимальный размер тома	32 MiB	2 GiB 4 GiB	8 TiB
Возможности			
Сохраняемые даты	Создания, модификации, доступа		
Диапазон дат	1 января, 1980 - 32 декабря, 2107 ^{источник?} ¹		
Дополнительные данные	Изначально не поддерживаются		
Attributes	Только для чтения, скрытый, системный, метка тома, подкаталог, архивный		
Разграничение прав доступа	Нет		
Прозрачное сжатие	Per-volume, Stacker , DoubleSpace , DriveSpace		Нет
Прозрачное шифрование	Per-volume only with DR-DOS		Нет

FAT (от [англ.](#) *File Allocation Table* — «таблица размещения файлов») — [файловая система](#), используемая в [операционных системах DOS](#) и [Windows](#).

Структура системы FAT

[Логический диск](#), отформатированный в системе FAT, имеет следующие разделы:

- [загрузочный сектор](#);
- таблица размещения [файлов](#) — собственно FAT (традиционно в двух экземплярах);
- корневой [каталог](#);
- файлы.

Для хранения файлов всё доступное для них пространство разбивается на [кластеры](#). Таблица размещения файлов содержит ячейки, каждая из которых указывает на определённый кластер на жестком диске. Если кластер принадлежит файлу, то его ячейка содержит номер следующей ячейки этого же файла. Если ячейка указывает на конец файла, она содержит значение «FFFF». Неиспользуемые кластеры помечены нулём. «Плохие» кластеры помечены специальным кодом.

Максимальный размер кластера, который поддерживается в FAT, составляет 32 [Кб](#). Зная, что максимальное количество кластеров, которое можно проадресовать шестнадцатиразрядным указателем равно 65536, можно вычислить какой величины раздел можно отформатировать, применяя тот или иной размер кластера. Если взять размер кластера равным размеру физического кластера(сектора), то получим: 65536*512=32 [Мб](#). Если взять кластер в 2 раза больше, то можно отформатировать раздел уже до 64 [Мб](#). Ввиду того, что разрядность [ФС](#) величина постоянная, для форматирования дисков различных размеров будут применяться разные размеры кластеров. Например, чтобы отформатировать диск более 1 [Гб](#), нужно применять кластер 32 [Кб](#). Поскольку такой размер кластера является максимально допустимым в этой ФС, то можно определить, что максимальный размер раздела, формируемый под FAT, равен 2 [Гб](#).

Отказоустойчивость системы

Поскольку система FAT хранит данные о файлах и данные о свободном месте на диске в одной таблице, то операция по записи файла, традиционно состоящая из двух этапов (добавление занимаемого блока в перечень занятых и исключение этого же блока из списка свободных) происходит в FAT в одно действие. Благодаря этому система FAT обладает **врождённой устойчивостью к сбоям**, то есть сбой (например, питания) в момент выполнения операции чтения или записи в большинстве случаев не приведёт к разрушению файловой системы. Однако следует помнить, что в данном случае речь идёт именно о целостности файловой системы, а не самих файлов.

Версии системы FAT

Существует три версии FAT — [FAT12](#), [FAT16](#) и [FAT32](#). Они отличаются количеством **бит**, отведённых для хранения номера кластера. FAT12 применяется в основном для [дискет](#), FAT16 — для дисков малого объёма.

В различных операционных системах также были внедрены различные расширения FAT. Например, в [DR-DOS](#) имеются дополнительные атрибуты доступа к файлам; в [Windows 95](#), [Linux](#) и [Proolix](#) — поддержка длинных имён файлов в формате Unicode (VFAT); в [OS/2](#) — расширенные атрибуты файлов.

NTFS

NTFS (от [англ.](#) *New Technology File System* — «файловая система новой технологии») — стандартная [файловая система](#) для семейства операционных систем [Microsoft Windows NT](#).

NTFS заменила использовавшуюся в [MS-DOS](#) и [Microsoft Windows](#) файловую систему [FAT](#). NTFS поддерживает систему [метаданных](#) и использует специализированные структуры данных для хранения информации о файлах для улучшения производительности, надёжности и эффективности использования дискового пространства. NTFS имеет встроенные возможности разграничивать доступ к данным для различных пользователей и групп пользователей, а также назначать квоты (ограничения на максимальный объём дискового пространства, занимаемый теми или иными пользователями). NTFS использует систему [журналирования](#) для повышения надёжности файловой системы.

NTFS разработана на основе файловой системы [HPFS](#) (от [англ.](#) *High Performance File System* - высокопроизводительная файловая система), создававшейся [Microsoft](#) совместно с [IBM](#) для операционной системы [OS/2](#). Но, получив такие несомненно полезные новшества как квотирование, журналируемость, разграничение доступа и аудит, в значительной степени утратила присущую прародительнице ([HPFS](#)) весьма высокую производительность файловых операций.

Различают несколько версий NTFS: v1.2 используется в [Windows NT 3.51](#) и [Windows NT 4.0](#), v3.0 поставляется с [Windows 2000](#), v3.1 — с [Windows XP](#) и [Windows Server 2003](#). Иногда последние версии обозначают как v4.0, v5.0 и v5.1 в соответствии с версиями Windows NT, с которыми они поставляются.

Windows NT имеет NTFS 4.0, а Windows 2000/XP - NTFS 5. При подключении диска с NTFS 4.0 к 2000/XP он автоматически и без предупреждений конвертируется системой в NTFS 5.0. [\[1\]](#)

Поддержка операционными системами

В данный момент полноценная поддержка NTFS присутствует только в ОС Microsoft семейства [Windows NT](#). Для других систем в данный момент существуют следующие средства доступа к NTFS-разделам:

- [MS-DOS](#)
 - Драйвер [NTFSDOS Марка Руссиновича](#) (Mark Russinovich) — поддержка чтения, а в версии [Professional](#) и записи на NTFS-разделы (с июля 2006 права на утилиты Марка Руссиновича

принадлежат Microsoft [1]).

- **Windows 9x**
 - Драйвер **NTFS for Windows 98** того же Марка Русиновича[1]. Поддерживает чтение с разделов, но для записи можно использовать **DOS'овский драйвер**.
 - Драйвер Paragon NTFS for Windows 98. Продукт коммерческий. Есть поддержка записи, но в демо-версии — только чтение. Однако, судя по отсутствию какой-либо информации на сайте производителя, продукт им больше не поддерживается.
- **GNU/Linux**
 - Проект **Linux-NTFS**. Включает модуль ядра, а также набор утилит для различных операций с файловыми системами NTFS (проверка целостности, восстановление удалённых файлов, изменение размера и др.). Именно этот драйвер включается в ядро **Linux** ещё с версии 2.2 (с тех пор он был практически полностью переписан). Модулем ядра поддерживается практически только чтение (запись — лишь в существующие файлы без изменения их размера), но недавно в рамках проекта появилась утилита ntfsmount, использующая FUSE и позволяющая монтировать NTFS-разделы на запись с некоторыми ограничениями. Это первый полностью свободный продукт, имеющий такую возможность.
 - Проект **NTFS-3G**. Возник как ответвление от предыдущего проекта, отличается более полной поддержкой записи на NTFS-разделы. Объявлен стабильным (выпущена версия 1.0) 21 февраля 2007 года. Обеспечивает высокий уровень надежности и производительности. В настоящее время проект активно развивается.
 - Проект **Captive NTFS** ([2]). Создавался как «обёртка» для использования оригинального драйвера Windows NT в ОС GNU/Linux. Имеется возможность записи, однако драйвер работает крайне медленно и имеет некоторые технические ограничения.
 - Драйвер Paragon **NTFS for Linux**. Поддерживается чтение и запись, также имеется ряд утилит для создания и некоторых операций с NTFS-разделами. Продукт платный.
- **KolibriOS**
 - Поддержка чтения разделов и запуска программ с версии 0.6.5. Примечательным фактом является то, что драйвер написан на **ассемблере**.

Сравнительная таблица файловых систем FAT и NTFS

Ограничения	NTFS	FAT16 и FAT 32
Размеры тома	Минимальный размер тома составляет приблизительно 10 Мб.	FAT поддерживает различные размеры томов — от объема дискет и до 4 Гб.
	На практике рекомендуется создавать тома, размеры которых не превышают 2 Тб.	FAT32 поддерживает тома объемом от 2 гб. до 2 Тб. Работая под управлением Windows XP для FAT32 можно отформатировать тома, размер которых не превышает 32 Гб.
	С помощью NTFS нельзя форматировать дискеты.	FAT32 не поддерживаются диски размеры которых меньше 512 Мб.
Размеры файлов	Теоретически размер файла может составлять 16 эксабайт .	FAT16 поддерживает файлы размером не более 2 Гб. FAT32 поддерживает файлы размером не более 4 Гб.

ext2

- **ext2** или **2я расширенная файловая система** — **файловая система** для **ядра Linux**. Она была разработана Rémy Card'ом в качестве замены для **extended file system**. Она достаточно быстра для того, что бы служить эталоном в **тестах производительности** файловых систем. Она не является **журналируемой файловой системой** и это её главный недостаток. Развитием ext2 стала журналируемая файловая система **ext3**, полностью совместимая с ext2.

История

На заре развития Linux использовала файловую систему [Minix](#). Эта файловая система была довольно стабильна, но была 16-разрядной и как следствие имела жесткое ограничение в 64 [Мегабайта](#) на раздел. Также присутствовало ограничение имени файла: оно составляло 14 символов. Эти и не только ограничения повлекли появление в апреле 1992 года «расширенной файловой системы» (extended file system), решавшей 2 главные проблемы Minix. Новая файловая система расширила ограничения на размер [файла](#) до 2 [гигабайт](#) и установила предельную длину [имени файла](#) в 255 символов. Но она все равно имела проблемы: не было поддержки раздельного доступа, временных меток модификации данных.

Решением всех проблем стала новая файловая система, разработанная в январе 1993 года. В ext2 были сразу реализованы соответствующие стандарту [POSIX](#) списки контроля доступа [ACL](#) и расширенные [атрибуты](#) файлов.

Логическая организация файловой системы ext2

Сетевая иерархия каталогов [файловой системы](#) ext2

[Граф](#), описывающий [иерархию каталогов](#), [файловой системы](#) ext2 представляет собой сеть, это достигается тем, что один [файл](#) может входить сразу в несколько каталогов.

Все типы файлов имеют символьные имена. В иерархически организованных файловых системах обычно используются три типа имен — файлов: простые, составные и относительные. Не является исключением и «вторая расширенная файловая система». Ограничения на простое имя состоят в том что, его длина не должна превышать 255 символов, а также в имени не должны присутствовать символ NUL и `'/'`. Ограничения на символ NUL связаны с представлением строк на языке [Си](#), а на символ `'/'` с тем, что он используется как разделительный символ между каталогами. Полное имя представляет собой цепочку простых символьных имен всех [каталогов](#), через которые проходит путь от корня до данного [файла](#). В [файловой системе](#) ext2 файл может входить в несколько каталогов, а значит, иметь несколько полных имен; здесь справедливо соответствие «один файл — много полных имен». В любом случае полное имя однозначно определяет файл.

[Атрибутами файловой системы](#) ext2 являются:

- Тип и права доступа к [файлу](#);
- Владелец, группа;
- Информация о разрешенных операциях доступа к файлу;
- Времена создания, последнего доступа, последнего изменения и время последнего удаления;
- текущий размер файла;
- тип файла;
 - обычный файл;
 - [каталог](#);
 - файл байт-ориентированного устройства;
 - файл блочно-ориентированного устройства;
 - [Сокет](#);
 - [именованный канал](#);
 - [символическая ссылка](#);
- число блоков, занимаемых файлом;
- [ACL](#)
- другие

[Атрибуты](#) файлов хранятся не в каталогах, как это сделано в ряде простых [файловых систем](#), а в специальных [таблицах](#). В результате каталог имеет очень простую структуру, состоящую всего из двух частей: номера индексного дескриптора и имени файла.

Физическая организация файловой системы ext2

Структура дискового раздела

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 — 1024, 2048 и 4096 байт. Размер блока указывается при создании файловой системы на разделе диска. Меньший размер блока позволяет экономить место на жестком диске, но также ограничивает максимальный размер файловой системы. Все блоки имеют порядковые номера. С целью уменьшения фрагментации и количества перемещений головок жесткого диска при чтении больших массивов данных блоки объединяются в группы блоков.

Базовым понятием файловой системы является индексный дескриптор (информационный узел), information node, или inode. Это специальная структура, которая содержит информацию об атрибутах и физическом расположении файла.

Обобщенная структурная схема ФС ext2

Каждая группа блоков имеет одинаковое строение. Суперблок — основной элемент файловой системы ext2. 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;
- размер индексного дескриптора;
- идентификатор файловой системы;

От целостности суперблока напрямую зависит работоспособность файловой системы. Операционная система создает несколько резервных копий суперблока для возможности его восстановления в случае повреждения. Описание группы блоков, представляет собой массив, содержащий общую информацию обо всех блоках раздела. Битовая карта блоков — это структура, каждый бит которой показывает, отведен ли соответствующий ему блок какому-либо файлу. Если бит равен 1, то блок занят. Аналогичную функцию выполняет битовая карта индексных дескрипторов, показывая какие именно индексные дескрипторы заняты, а какие нет.

Все оставшееся место, обозначенное в таблице, как данные, отводится для хранения файлов.

Система адресации данных

Система адресации ФС ext2

Система адресации данных — это одна из самых существенных составных частей файловой системы. Именно система адресации позволяет находить нужный файл среди множества как пустых, так и занятых блоков на диске. Файловая система ext2 использует следующую схему адресации блоков файла. Для хранения адреса файла выделено 15 полей, каждое из которых состоит из 4 байт. Если размер файла меньше или равен 12 блокам, то номера этих кластеров непосредственно перечисляются в первых двенадцати полях

4.15

адреса. Если размер файла превышает 12 блоков, то следующее 13-е поле содержит адрес кластера, в котором могут быть расположены номера следующих блоков файла. Таким образом, 13-й элемент адреса используется для косвенной адресации. При максимальном размере блока равном 4096 байт, 13-й элемент, может содержать до 1024 номеров следующих кластеров данных файла. Если размер файла превышает 12+1024 блоков, то используется 14-е поле, в котором находится номер блока, содержащего 1024 номеров блоков, каждый из которых хранят 1024 номеров блоков данных файла. Здесь применяется уже двойная косвенная адресация. И наконец, если файл включает более $12+1024+1048576 = 1049612$ блоков, то используется последнее 15-е поле для тройной косвенной адресации.

Таким образом, описанная выше система адресации, позволяет при максимальном размере блока 4 Кб иметь файлы размера до 2 терабайт.