

WYOS - Выпуск №6

Рассмотрение NTFS.

commrade, Среда, 30 Июнь 2004, 01:30

Напиши свою ОС! #6

Здравствуйте, уважаемые подписчики!

В этом выпуске мы продолжим рассматривать различные файловые системы. Сегодня мы рассмотрим файловую систему NTFS.

NTFS

NTFS выросла из файловой системы HPFS, разрабатываемой совместно IBM и Microsoft для проекта OS/2. Она начала использоваться вместе с Windows NT 3.1 в 1993 году. Windows NT 3.1 должна была составить конкуренцию серверам на базе NetWare и Unix, поэтому NTFS вобрала в себя все тогдашние технологические достижения. Вот основные из них:

Работа с большими . NTFS имеет размер кластера 512 байт, что в принципе оптимально, но его можно менять до 64K. Более важно то, что NTFS способна теоретически работать с томами размером в 16,777,216 терабайт.

Теоретически, потому что таких жестких дисков пока просто не существует, и появятся они весьма не скоро.

. NTFS содержит две копии аналога FAT, которые называются MFT (Master File Table). В отличие от FAT MSDOS, MFT больше напоминает таблицу базы данных. Если оригинал MFT повреждён в случае аппаратной ошибки (например, появления bad-сектора), то система при следующей загрузке использует копию MFT, и автоматически создаёт новый оригинал, уже с учётом повреждений. Но это не самое главное. Главное, что NTFS использует систему транзакций при записи файлов на диск. Эта система пришла из СУБД, где защита целостности данных - жизненно важное дело. Уже это говорит о её эффективности. В упрощённом виде она работает так:

<BLOCKQUOTE>

Драйвер ввода/вывода NTFS инициирует процесс записи, одновременно сообщая сервису Log File Service вести лог всего происходящего.

Данные пишутся в кэш, под управлением сервиса Cache Manager.

Cache Manager посылает данные Virtual Memory Manager-у (менеджеру виртуальной памяти), для записи на диск в фоновом режиме.

Virtual Memory Manager посылает данные драйверу диска, пропустив их через Fault Tolerant Driver (если у вас массив дисков RAID).

Драйвер диска шлёт их контроллеру, который уже пишет их либо в кэш, либо прямо на диск.

Если эта операция проходит без ошибок, запись лога удаляется.

Если происходит сбой, запись лога остается в таблице транзакций, и при следующем доступе к диску Log File Service обнаруживает эту запись, и просто

<http://www.osrc.info/plugins/content/content.php?content.10>
восстанавливается как было до этой операции. Такая система

гарантирует абсолютную сохранность данных в случае копирования, перемещения и удаления файлов или директорий. При внесении изменений в файл, вы теряете те изменения, которые находились в момент сбоя в памяти или в кэше контроллера, и не успели записаться на диск. </BLOCKQUOTE>

<LI value=3>. NTFS рассматривает файлы, как объекты. Каждый файловый объект обладает свойствами, такими как его имя, дата создания, дата последнего обновления, архивный статус, и дескриптор безопасности. Файловый объект также содержит набор методов, которые позволяют с ним работать, такие как open, close, read и write. Пользователи, включая сетевых, для обращения к файлу вызывают эти методы, а Security Reference Monitor определяет, имеет ли пользователь необходимые права для вызова какого-либо из этих методов. Кроме этого, файлы можно шифровать. Правда, с шифрованием стоит быть осторожнее. Если у вас рухнет система, или вы её переустановите то вы не сможете прочитать зашифрованные файлы, если не имеете ERD.

Компрессия . NTFS позволяет компрессировать отдельные каталоги и файлы, в отличие от DriveSpace, который позволял сжимать только диски целиком. Это очень удобно, для экономии пространства на диске, например можно сжимать "на " большие графические файлы формата BMP, или текстовые файлы, причём для пользователя всё это будет прозрачно.

Поддержка формата ISO Unicode. Формат Unicode использует 16bit для кодировки каждого символа, в отличие от ASCII, который использовал 8bit, или ещё хуже - 7bit. Для простого пользователя это означает то, что теперь он может называть файлы на любом языке, хоть на китайском - система это будет поддерживать, не требуя изменить кодовую страницу, как это делал DOS и W9x.

<p>NTFS также включает возможности безопасности, требуемые для файловых серверов и высококачественных персональных компьютеров в корпоративной среде. NTFS поддерживает управление доступом к данным и привилегии владельца, что является важным для целостности корпоративных данных.

В то время как каталогам, разделяемым при помощи Windows NT Server, назначаются специфические разрешения, файлам и каталогам NTFS могут назначаться разрешения вне зависимости, разделены они или нет. NTFS — единственная файловая система в Windows NT, которая позволяет назначить разрешения для отдельных файлов.

Основное отличие данной файловой системы от FAT32 это то, что вся информация на томе NTFS является файлом или частью файла. Каждый распределенный на томе NTFS сектор принадлежит некоторому файлу. Даже метаданные (metadata) файловой системы (информация,

которая описывает непосредственно файловую систему) являются частью файла.

Эта основанная на атрибутах файловая система поддерживает

объектно-ориентированные приложения,

обрабатывая все файлы как объекты, которые имеют определяемые пользователем и системой атрибуты.

</p>

<p>Основные моменты, которые в дальнейшем будут рассмотрены это:

Главная файловая таблица.

Атрибуты файла NTFS.

Особенности NTFS.

</p>

Главная файловая таблица

<P>Каждый файл на томе NTFS представлен записью в специальном файле, называемом главной файловой таблицей (MFA — master file table). NTFS резервирует первые 16 записей таблицы для специальной информации. Первая запись этой таблицы описывает непосредственно главную файловую таблицу;</P>

<P>за ней следует зеркальная запись (mirror record)

MFT. Если первая запись MFT разрушена, то NTFS читает вторую запись для отыскания зеркального файла MFT, первая запись которого идентична первой записи MFT. Местоположения сегментов данных MFT и зеркального файла MFT записаны в секторе начальной загрузки. Дубликат сектора начальной загрузки находится в логическом центре диска.</P>

<P>Третья запись MFT — файл регистрации (log file);

используется для восстановления файлов. Файл регистрации подробно описан в настоящей главе ниже. Семнадцатая и последующие записи главной файловой таблицы используются собственно файлами и каталогами (также рассматриваются как файлы NTFS) на томе. На рисунке ниже показана упрощенная структура MFT:</P>

<P align=center></P>

<P>Главная файловая таблица отводит определенное

количество пространства для каждой записи файла. Атрибуты файла записываются в распределенное пространство MFT. Небольшие файлы и каталоги (обычно до 1500 байт или меньше), типа файла, показанного ниже, могут полностью содержать внутри записи главной файловой таблицы.</P>

<TABLE cellSpacing=0 cellPadding=2 border=1>

<TBODY>

<TR>

<TD vAlign=top width=13% height=53>Стандартная информация</TD>

<TD vAlign=top width=16% height=53>Имя файла или </TD>

<TD vAlign=top width=15% height=53>Дескриптор </TD>

<TD vAlign=top width=43% height=53>Данные или </TD>

<TD vAlign=top width=13% height=53> <P> </P></TD>

</TR>

</TBODY>

</TABLE>

<P>Подобный подход обеспечивает очень быстрый доступ к

файлам. Рассмотрим, например, файловую систему FAT, которая использует таблицу размещения файлов, в которой перечисляются имена и адрес каждого файла.

Элементы каталога FAT содержат индекс в таблице размещения файла. В случае если необходимо просмотреть содержимое файла, FAT

сначала читает таблицу размещения файлов и убеждается в существовании файла.

Далее FAT восстанавливает файл, ища цепочку распределенных блоков, относящихся

его использования.

Записи каталога помещены внутри главной файловой таблицы так же, как записи файла. Вместо данных каталоги содержат индексную информацию. Небольшие записи каталогов находятся полностью внутри структуры MFT. Большие каталоги организованы в B-tree, имея записи с указателями на внешние кластеры, содержащие элементы каталога, которые не могли быть записаны внутри структуры MFT.

Атрибуты файла NTFS

NTFS просматривает каждый файл (или каталог) как набор атрибутов файла. Такие элементы, как имя файла, информация зашиты и даже данные — все это атрибуты файла. Каждый атрибут идентифицирован кодом типа атрибута и, необязательно, именем атрибута.

Если атрибуты файла могут находиться внутри записи файла MFT, они называются резидентными (resident) атрибутами. Например, информация типа имени файла и отметки времени всегда включается в запись файла MFT. Если файл слишком большой, чтобы содержать все атрибуты в записи файла MFT, часть атрибутов является нерезидентной (nonresident). Нерезидентные атрибуты занимают один или несколько пробегов (run) дискового пространства в другом месте тома (пробег дискового пространства — непрерывная линейная область на диске).

Вообще, все атрибуты могут быть вызваны как поток бантов независимо от того, являются ли они резидентными или нерезидентными.

В таблице 1 представлен список всех атрибутов файла, в настоящее время определенных для NTFS. Этот список расширяем, т. е. другие атрибуты файла в будущем могут быть определены в случае необходимости.

Таблица 1: Атрибуты файла NTFS

Тип

--

Standard Information (стандартная информация)

Включает бюджет связи и так

Attribute List (список атрибутов)

Перечисляет все другие атрибуты (только в больших файлах)

<TD vAlign=top>Атрибут, повторяющийся для длинных и для коротких имен файлов Длинное имя файла может содержать до 255 символов Unicode Короткое имя — доступно для MS-DOS, восемь плюс три символа, без учета регистра Дополнительные имена, или жесткие связи (hard links), используются POSIX и могут быть также включены в качестве дополнительных атрибутов имени </TD>

</TR>

| <TR> | |
 <TD vAlign=top>Security Descriptor (дескриптор безопасности)</TD> |

<p><TD vAlign=top>Фиксирует информацию о том, кто может обращаться к файлу, кто является его владельцем и так</p>	
---	--

</TD>

</TR>

| | |
 <TD vAlign=top>Data (данные)</TD> | <TD vAlign=top>Содержит данные </TD> || <TR> | |
 <TD vAlign=top>Index Root (корень индексов)</TD> | `<TD vAlign=top>Используется при работе с </TD>` || | |

<TD vAlign=top>Index Allocation (индексное размещение)</TD>

 <TD vAlign=top>Используется при работе с </TD> |

</TR>

| | |
 <TD vAlign=top>Volume Information (информация тома)</TD> |

Используется только в системном файле тома и включает в частности версию и имя
--

| | |
 <TD vAlign=top>Bitmap (битовый массив)</TD> |

<TD vAlign=top>Предоставляет информацию об использовании записей в MFT или </TD>

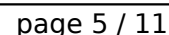
<TR>

Extended Attribute Information(информация расширенного атрибута)
--

Используется файловыми серверами, которые связаны с системами OS/2 Этот тип атрибута не используется Windows NT

| | <TR> |

<TD vAlign=top>Extended Attributes (расширенные



атрибуты)</TD>

<TD vAlign=top>Используется файловыми серверами, которые связаны с системами OS/2 Этот тип атрибута не используется Windows NT</TD>

</TR>

</TBODY>

</TABLE>

Особенности NTFS

<p>Итак какие же особенности NTFS следует рассмотреть:

Возможность работать с длинными и короткими

Управление потоками

Согласованность с POSIX

Возможности NTFS, используемые Macintosh Services Clients

Представление служебной информации NTFS в виде

</p>

<P>Длинные и короткие имена </P>

<P>Подобно HPFS, NTFS поддерживает имена файла до 255

символов. Имена файла NTFS используют набор символов Unicode с 16 битами;

однако вопрос доступа из MS-DOS решен. NTFS автоматически генерирует

поддерживаемое MS-DOS имя (восемь плюс три символа) для каждого файла. Таким

образом, файлы NTFS могут использоваться через сеть операционными системами

MS-DOS и OS/2. Это особенно важно для файловых серверов организации, которая

использует персональные компьютеры с двумя или всеми тремя этими операционными системами.</P>

<P>Создавая имена файла “восемь плюс три”, NTFS также

позволяет приложениям MS-DOS и Windows 3-х работать с файлами, имеющими

длинные имена NTFS Кроме того, при сохранении файла приложениями MS-DOS или

Windows 3. на томе NTFS сохраняются и имя файла “восемь плюс три” и длинное

имя NTFS.</P>

<P>Внимание! При сохранении файла приложениями MS-DOS

или Windows 3.x на томе NTFS, если приложение сохраняет временный файл,

удаляет первоначальный файл и переименовывает временный файл с первоначальным

именем, длинное имя файла теряется. Любой уникальный набор разрешений файла

также теряется. Разрешения передаются заново из родительского

каталога.</P>

<P>Если длинное имя файла или каталога на томе NTFS

содержит пробелы, следует убедиться, что путь записан в кавычках.</P>

<P>Ниже приведен пример легальной записи командной

строки, содержащей пробелы:</P>

<P><code>move "c:\This month's reports*.*" "c:\Last month's reports"</code></P>

<P>Внимание! Необходимо внимательно подходить к

использованию групповых символов типа * и ? вместе с командами del и copy. При

выполнении этих команд NTFS работает и с длинным и с коротким именем файла;

таким образом могут быть удалены или скопированы лишние файлы.</P>

чувствительным к регистру длинными именами, самым надежным способом является выбор файлов с использованием мыши в File Manager. Этот способ позволяет однозначно определить файлы для выполнения операций над ними.

для имен файлов, существует возможность задействования нескольких

Потоки

<P>Следующий пример иллюстрирует один из

ПОТОКОВ:

<P><code>myfile.dat:stream2</code></P>

<P>Эта возможность позволяет управлять связующими

данными как отдельным модулем. Например, компьютеры Macintosh используют этот тип структуры для управления ветвлениями данных и ресурсами. Другим примером может служить библиотека файлов, в которой файлы определены как альтернативные потоки:

<P><code>library: file1 : file2 : file3</code></P>

<P>Или предположим, что “интеллектуальны” транслятор создает структур файла подобно следующей:

<P><code>program: source file : doc file : object file : executable file</code></P>

Внимание! Т. к. NTFS не поддерживается на гибких

дисках, то при копировании файлов NTFS на гибкий диск потоки данных и другие не обеспечиваемые FAT атрибуты теряются.

Согласованность с POSIX

<P>Согласованностьс POSIX позволяет переносить

приложения UNIX в среду Windows NT. Windows NT полностью согласована со стандартом 1003.1 института IEEE, который определяет присвоение имен и идентификацию файлов.

Следующие возможности POSIX включены в NTFS:

<u|>

- Чувствительные к регистру** Для POSIX файлы README.TXT, Readme.txt и readme.txt являются различными.

Жесткие связи (hard links) Файлу может быть

присвоено несколько имен. Это позволяет двум файлам с различными именами,

данные.

Дополнительные отметки

Показывают, когда файл был последний раз использован или изменен.

Внимание! Несмотря на то что NTFS поддерживает регистрозависимые имена, нельзя использовать стандартные операции NTFS для управления файлами, имена которых отличаются только регистром (к стандартным операциям относятся выполняемые из командной строки, типа copy, del и move, и их эквиваленты в File Manager). Например, оба файла annm.doc и AnnM.Doc будут удалены при использовании следующей команды:

```
del AnnM.Doc
```

Таким образом, для управления файлами, имена которых отличаются только регистром, следует использовать приложения POSIX.

Возможности NTFS, используемые Macintosh Services Clients

Сервис для Macintosh входит в состав Windows NT

Server. Этот сервис предоставляет пользователям Macintosh возможность доступа к файлам, находящимся на Windows NT Server; т. к. эти файлы доступны сетевым пользователям Windows NT, файловый сервер может быть легко использован для общего доступа с различных аппаратных платформ.

При разрешении сервиса для Macintosh следует сделать доступным раздел NTFS, если требуется автоматическое создание тома модуля аутентификации пользователя (User Authentication Module) для клиентов Macintosh (Network Control Panel использует первый раздел NTFS для создания этих томов по умолчанию).

Клиенты Macintosh могут использовать только файлы на томах NTFS. Ветвления ресурсов Macintosh и информация Finger для каждого файла Macintosh сохраняются как потоки NTFS. Поскольку NTFS поддерживает длинные имена, большинство имен файлов Macintosh сохраняются. Сервис для Macintosh сохраняет привилегии папки (File Sharing folder) как разрешения Windows NT; это означает, что существует только один набор разрешений для папки или файла, который предписан и пользователям Windows NT и пользователям Macintosh. Однако пользователи Macintosh не смогут видеть разрешения файла, так как AppleShare поддерживает только разрешения папки.

Системные файлы NTFS

NTFS включает несколько системных файлов, которые скрыты от просмотра на томе. Системные файлы используются только файловой системой для хранения метаданных и поддержания работы файловой системы. Системные файлы записываются на том утилитой Format. Список системных файлов NTFS представлен в таблице 2.

Таблица 2: Системные файлы NTFS

--

--

--

Системный

```

<TD vAlign=top width=16% height=20><B><FONT face=Arial size=2>Имя
</FONT></B></TD>
<TD vAlign=top width=48% height=20><B><FONT face=Arial
size=2></FONT></B></TD></TR>
<TR>
<TD vAlign=top width=36%><FONT face=Arial size=2>Master File
Table</FONT>
<P><FONT face=Arial size=2>(главная файловая таблица)</FONT></P></TD>
<TD vAlign=top width=16%><FONT face=Arial size=2>□Mft</FONT>
<P>&nbsp;</P>
<P>&nbsp;</P></TD>
<TD vAlign=top width=48%><FONT face=Arial size=2>Список содержимого
тома NTFS</FONT>
<P>&nbsp;</P>
<P>&nbsp;</P></TD></TR>
<TR>
<TD vAlign=top width=36% height=78><FONT face=Arial size=2>Master File
Table2</FONT>
<P><FONT face=Arial size=2>(копия главной файловой </FONT></P></TD>
<TD vAlign=top width=16% height=78><FONT face=Arial
size=2>□MftMirr</FONT>
<P><FONT face=Arial size=2>-)</FONT></P></TD>
<TD vAlign=top width=48% height=78><FONT face=Arial size=2>Зеркальное
отображение наиболее </FONT>
<P><FONT face=Arial size=2>частей MFT, используется для </FONT></P>
<P><FONT face=Arial size=2>рантирования доступа к MFT в случае
сбоя</FONT></P>
<P><FONT face=Arial size=2>одиночного </FONT></P></TD></TR>
<TR>
<TD vAlign=top width=36% height=46><FONT face=Arial size=2>Log
File</FONT>
<P><FONT face=Arial size=2>(регистрационный файл)</FONT></P></TD>
<TD vAlign=top width=16% height=46><FONT face=Arial
size=2>□LogFile</FONT>
<P>&nbsp;</P></TD>
<TD vAlign=top width=48% height=46><FONT face=Arial size=2>Список
шагов транзакции </FONT>
<P><FONT face=Arial size=2>Log File System для восстановления состо яния
(recoverability)</FONT></P></TD></TR>
<TR>
<TD vAlign=top width=36%><FONT face=Arial size=2>Volume
(том)</FONT></TD>
<TD vAlign=top width=16%><FONT face=Arial size=2>□Volume</FONT></TD>
<TD vAlign=top width=48%><FONT face=Arial size=2>Имя версия и другая
информация относительно </FONT></TD></TR>
<TR>
<TD vAlign=top width=36%><FONT face=Arial size=2>Attribute
Definition</FONT>

```

<P>(определение атрибутов)</P></TD>

<TD vAlign=top width=16%>□AttrDef

<P> </P>

<P> </P></TD>

<TD vAlign=top width=48%>Таблица имен атрибутов номеров

<P></P></TD></TR>

<TR>

<TD vAlign=top width=36%>Root Filename Index (индекс корня файловых имен)</TD>

<TD vAlign=top width=16%>□</TD>

<TD vAlign=top width=48%>Корневой </TD></TR>

<TR>

<TD vAlign=top width=36% height=47>Cluster Bitmap

<P>(битовый массив кластеров)</P></TD>

<TD vAlign=top width=16% height=47>□Bitmap

<P> </P>

<P> </P></TD>

<TD vAlign=top width=48% height=47>Описание содержимого тома показываю

<P>щее какие размещаемые модули </P></TD></TR>

<TR>

<TD vAlign=top width=36% height=48>Boot File

<P>(загрузочный файл)</P></TD>

<TD vAlign=top width=16% height=48>□Boot

<P> </P>

<P> </P></TD>

<TD vAlign=top width=48% height=48>Содержит информацию начальной

<P>для тома если том является </P></TD></TR>

<TR>

<TD vAlign=top width=36% height=48>Bad Cluster File

<P>(файл плохих кластеров)</P></TD>

<TD vAlign=top width=16% height=48>□BadClus

<P> </P>

<P> </P></TD>

<TD vAlign=top width=48% height=48>Содержит указание положения плохих кластеров </TD></TR></TBODY></TABLE>

<P> </P>

Заключение

<p>В настоящее время в последних версиях операционных систем Windows 2000 и Windows XP

используется последняя на сегодняшний день версия данной файловой системы NTFS5.

Что нового в файловой системе NTFS5?

Самое главное, за что ругали NT4, и в чём она уступала NetWare, это за отсутствие квотирования.

Квотирование - это ограничение максимального объёма дискового пространства для пользователя,

которое он сможет использовать. Зачем это нужно? Ну, на компьютерах, работающих в больших сетях,

в фирмах и организациях и без вопросов понятно зачем, ведь жесткие диски не резиновые.

А на домашнем компьютере, можно например, выделить пользователю "" или "младший " не более 50МБ на винчестере, чтобы они не захламляли диск чем попало. Причём, вовсе не обязательно, чтобы все файлы пользователя хранились в одном месте, они могут быть разбросаны

по всем дискам.</p>

Ссылки

<p>http://www.emanual.ru - библиотека электронной</p>