

10 – Sistemas de Arquivos

SCC0503 – Algoritmos e Estruturas de Dados II

Prof. Moacir Ponti Jr.
www.icmc.usp.br/~moacir

Instituto de Ciências Matemáticas e de Computação – USP

2011/1

- 1 Introdução
 - Tipos de sistemas de arquivos
- 2 Sistemas Operacionais e Sistemas de Arquivos
- 3 Características de alguns sistemas
 - FAT
 - NTFS
 - ext
 - UDF
 - Novos sistemas

- 1 Introdução
 - Tipos de sistemas de arquivos
- 2 Sistemas Operacionais e Sistemas de Arquivos
- 3 Características de alguns sistemas
 - FAT
 - NTFS
 - ext
 - UDF
 - Novos sistemas

- Método para armazenar, organizar e recuperar coleções de dados, de forma que seja facilmente gerenciável por humanos.
- São usados em dispositivos de armazenamento externo, incluindo HDs, CDs, DVDs, Pendrives e outros.
- Podem prover acesso aos dados num servidor de arquivos agindo como clientes num protocolo de rede (NFS, SMB)
- Endereçam os dados em unidades fixas de **clusters** ou **blocos**.
- Sistemas operacionais antigos eram basicamente sistemas de gerenciamento de arquivos (DOS: disk operating system).

- **Nomes de arquivos:** nome atribuído a um arquivo físico, usualmente ligando um nome do arquivo a um índice numa tabela ou outra estrutura de dados.
- **Metadados:** informação associada a arquivos mas que não fazem parte dos dados do arquivo (tamanho, número de blocos alocados, data e hora da última modificação).

Tipos de sistemas de arquivos

- **Discos:** geralmente baseado em “journaling”, um método que mantém as mudanças num disco em um log circular antes de realizar a operação no disco: FAT, NTFS, ext3/4, ReiserFS/4.
- **Discos óticos:** geralmente permitem atualizações incrementais e escrita em uma única passada: UDF (universal disk format).
- **Flash:** apesar de ser possível usar alguns sistemas de arquivos de disco, sistemas específicos foram criados para lidar com as características do dispositivo: JFFS2, YAFFS.
- **Fita:** diretório de arquivos espalhados pela fita: LTFS IBM.

- 1 Introdução
 - Tipos de sistemas de arquivos
- 2 Sistemas Operacionais e Sistemas de Arquivos
- 3 Características de alguns sistemas
 - FAT
 - NTFS
 - ext
 - UDF
 - Novos sistemas

Num sistema de arquivos plano (flat file system), não há subdivisão em diretórios ou pastas, todos os dados são armazenados num mesmo nível.

- Simples, tornou-se inviável conforme o número de arquivos cresceu,
- Usado em sistemas pequenos até o Apple Macintosh (1984).
- É utilizado pelo serviço de armazenamento web Amazon S3, que inclui web hosting, backup e image hosting.

Sistema de arquivos virtual com uma hierarquia a partir de uma raiz (*root*)

- Para acessar um dispositivo de armazenamento secundário é preciso definir um diretório no qual os arquivos desse dispositivo irão aparecer
- É chamado de montagem o processo de ativar um dispositivo e atribuir a ele um diretório (ponto de montagem).
- Quando o sistema inicia (boot), a montagem de alguns dispositivos pode ser feita utilizando uma tabela com configurações (*fstab*)
- Nos sistemas mais recentes existem detectores de mídia (CD, DVD, USB drives)
- Sistemas comuns: *ext** (*ext3*, *ext4*), *XFS*, *JFS*, *btrfs* e *ReiserFS*.

Derivados do Unix, possuem sistemas de arquivos distintos

- Solaris: VxFS, ZFS,
- Mac OS X: HFS Plus, UFS.

Usa uma abstração em letra para que o usuário possa distinguir entre um disco e outro (ou uma partição e outra).

- Derivado do sistema DOS, que reservava A e B para disquetes, e o C foi usado posteriormente para o disco rígido.
- Os arquivos de cada disco ou partição são organizados a partir de um diretório raiz correspondente à sua letra.
- Drives de rede podem ser mapeados para letras.
- Inicialmente foi desenvolvido o sistema FAT (12, 16, 32 e exFAT).
- Atualmente NTFS é utilizado como padrão.

FAT — File Allocation Table

- Técnica simples, suportada por quase todos os sistemas,
- Conveniente para compartilhar dados entre sistemas operacionais,
- Primeira implementação usava uma tabela com elementos de 8 bits, depois estendido no DOS para 12, e depois para 16 e 32.
- Usa uma tabela que centraliza a informação sobre as áreas livres do disco e a localização física de cada arquivo.
- O espaço no disco é alocado em grupos contíguos de *clusters*.
- Não há mecanismos para prevenir que novos arquivos fiquem fragmentados.

FAT12

- projetado para disquetes
- endereçamento por 12 bits: número de clusters = 4078
- gerenciamento difícil pois os registradores possuíam 8 ou 16 bits.
- um disquete típico, na época tinha 40 trilhas e 8 setores por trilha (320 setores de 512B = 160kB), tornando o FAT12 mais que suficiente, sendo usado até nos disquetes de 1.44MB.

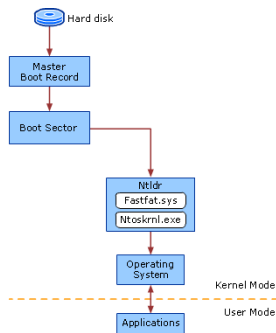
FAT16

- limite de 64 setores por cluster (cada cluster 32kB)
- na época (1988) fixou o limite “definitivo” de 2 GB por partição

FAT32

- clusters representados por números de 32 bits, sendo 28 usados para o número do cluster (268 milhões de clusters).
- contagem dos setores também é feita por 32 bits, limitando o volume a 2TiB.
- limite de tamanho de arquivo é de $2^{32} - 1$ bytes.

FAT — File Allocation Table



- **Boot sector:** partição iniciável com informação sobre o formato do volume, a estrutura do sistema de arquivos, e carrega Ntldr.dll
- **Master Boot Record:** contém um executável que carrega a BIOS para a RAM e verifica a tabela de partições.
- Ntldr.dll: inicia o sistema de arquivos e lê o conteúdo do arquivo Boot.ini com as configurações de inicialização.
- Fastfat.sys: driver para FAT16 e FAT32.
- Ntoskrnl.exe: extrai informação sobre qual drive carregar e a ordem.
- **Kernel Mode:** modo que permite acesso ao hardware.
- **User Mode:** modo no qual os aplicativos rodam.

FAT — File Allocation Table

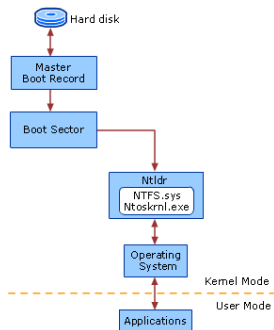
Organização de um volume FAT

Boot	Reserved	FAT1	FAT2	Root	Other folders	
sector	sectors			Folder	and all files	

- **Reserved sectors:** setores que precedem o início da primeira FAT, incluindo o boot sector.
- **FAT1:** FAT original.
- **FAT2:** Cópia de segurança da FAT.
- **Root folder:** descreve os arquivos e pastas na raiz da partição
- **Other folder and all files:** contém dos dados para os arquivos e pastas dentro do sistema de arquivos.

- Criado para suceder o FAT32 e solucionar alguns problemas do antigo sistema
- Algumas das funcionalidades incluídas foram:
 - Rastreamento de links
 - Gerenciamento de arquivos esparsos
 - Journaling: log persistente de mudanças realizadas num volume
 - Hard links
 - Recuperação de estados anteriores (*snapshot*)

NTFS — new technology file system



- **Boot sector:** partição iniciável com informação sobre o formato do volume, a estrutura do sistema de arquivos, e carrega Ntldr.dll
- **Master Boot Record:** contém um executável que carrega a BIOS para a RAM e verifica a tabela de partições.
- Ntldr.dll: inicia o sistema de arquivos e lê o conteúdo do arquivo Boot.ini com as configurações de inicialização.
- Ntfs.sys: driver para NTFS.
- Ntoskrnl.exe: extrai informação sobre qual drive carregar e a ordem.
- **Kernel Mode:** modo que permite acesso ao hardware.
- **User Mode:** modo no qual os aplicativos rodam.

NTFS — new technology file system

- Permite clusters de tamanho entre 512bytes e 4KB, a depender do tamanho do volume.
- Em discos de 2GB a 2TB o cluster padrão é de 4KB.
 - É possível definir o tamanho do cluster em até 64KB no momento da formatação (mas não suporta a compressão de arquivos esparsos).
- O tamanho teórico máximo de um volume NTFS é de $2^{64}-1$ clusters. Usando o tamanho padrão são 16TB-4KB.
- Máximo de arquivos por volume: $2^{32} - 1 = 4,3$ bilhões de arquivos.
- Tamanho máximo de um arquivo: $2^{44} - 64KB = 16 TB$.

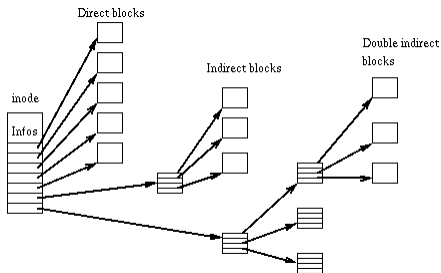
NTFS — new technology file system

Organização de um volume NTFS

NTFS	Master	File	Master File
Boot Sector	File Table	System Data	Table Copy

- **NTFS Boot Sector:** bloco BIOS com informação sobre o volume, a estrutura do sistema de arquivos e o carregador do SO.
- **Master File Table:** informação para recuperação de arquivos na partição NTFS.
- **File System Data:** dados não contidos na MFT
- **Master File Table Copy:** cópia de segurança dos registros para a recuperação de arquivos.

- Divide o espaço do volume em blocos.
- Os blocos são divididos em grupos de blocos.
- Os dados de um determinado arquivo são tipicamente escritos num único grupo de blocos quando possível para reduzir a fragmentação e minimizar o número de seeks quando uma quantidade grande de dados é lida.



- Não usa tabela, mas inodes. Cada arquivo ou diretório é representado por um inode, que inclui dados sobre o tamanho, permissão, propriedade (ownership) e localização no disco.
- O superbloco contém informação para a inicialização do sistema operacional.
- O diretório raiz (*root*) é armazenado no inode dois
- Cada inode contém as informações e ponteiros para os blocos seguintes (diretos e indiretos).

- O **ext3** é uma extensão do ext2 com: *journaling*, crescimento online do sistema de arquivos e indexação Htree para diretórios maiores.

Journaling

Salva as mudanças a serem feitas no disco em um log (*journal*) numa área dedicada do sistema de arquivos, antes de realizar a escrita no disco.

- Facilita a recuperação quando há uma falha. Após uma falha o sistema lê o *journal* e realiza as modificações até que o sistema esteja consistente.
- Por exemplo, ao deletar um arquivo o sistema:
 - 1 Remove a entrada de diretório
 - 2 Marca como livre o espaço para o arquivo e seu inode.

Se um problema ocorre entre os passos 1 e 2, haverá um inode órfão. Se apenas o passo 2 é realizado, o arquivo ainda não deletado será marcado como livre.

- Os limites para um sistema ext3 é de 2^{32} blocos. Como os blocos são tipicamente de 4KB, o tamanho máximo de um arquivo é $\approx 2\text{TiB}$ e do volume de $\approx 16\text{TiB}$.
- Algumas desvantagens do sistema ext3:
 - a inexistência de uma ferramenta de defragmentação,
 - a dificuldade de recuperar arquivos deletados,
 - não possuir suporte à recuperação de estados anteriores do sistema (snapshot),
 - não realizar verificação (*checksum*) do *journal* .

- O **ext4** é o mais recente sistema da série ext
- Funcionalidades adicionadas incluem:
 - Suporte a volumes de até 1EB e arquivos de até 16TB.
 - Uso de extents (um único extent pode ter até 128MB de espaço contíguo) e 4 extents por inode.
 - Pre-alocação de espaço persistente com preferência por espaços contíguos.
 - Verificação (*checksum*) do *journal*.

- Implementação utilizada por discos óticos.
- Projetado para permitir atualizações incrementais em discos graváveis e re-graváveis.
- Normalmente, os softwares escrevem em discos óticos um sistema de arquivo UDF em um único passo.
- O uso de múltiplas sessões também é permitido, mas gera muitas vezes problemas de leitura.
- Há 3 tipos principais
 - **Plain**: similar a um FAT, permite apenas escrita em uma passada.
 - **VAT** (virtual allocation table): insere uma estrutura adicional para remapear blocos quando arquivos ou outros dados são modificados. O espaço de arquivos deletados não pode ser recuperado.
 - **Spared** (RW): setores podem ser reescritos aleatoriamente, e o disco apagado. Mantém uma tabela (*sparing table*) com os setores defeituosos e um remapeamento para setores normais.

- Reiser4: sucessor do ReiserFS.
- LogFS: projetado para dispositivos de memória flash com grande capacidade.



FOLK, M.J. et al

File Structures: an object-oriented approach with C++
Capítulos 1 e 2



FOLK, M.J. et al

File Structures
Capítulos 1 e 2



YOUNG, J.H.

File Structures

<http://www.comsci.us/fs/notes/ch01a.html>



Microsoft TechNet

How NTFS Works

[http://technet.microsoft.com/en-us/library/cc781134\(W.S.10\).aspx](http://technet.microsoft.com/en-us/library/cc781134(W.S.10).aspx)