

Aula 9: Sistema de Arquivos Parte II

Professor(a): Virgínia Fernandes Mota

<http://www.dcc.ufmg.br/~virginiaferm>

OCS (TEORIA) - SETOR DE INFORMÁTICA



Implementando o Sistema de arquivos - Diretórios

- Quando um arquivo é aberto, o Sistema Operacional utiliza o caminho fornecido pelo usuário para localizar o diretório de entrada;
- O diretório de entrada provê as informações necessárias para encontrar os blocos no disco nos quais o arquivo está armazenado:
 - Endereço do arquivo inteiro (alocação contínua);
 - Número do primeiro bloco do arquivo (alocação com listas encadeadas);
 - Número do i-node;
- O serviço de diretório é responsável por mapear o nome ASCII do arquivo na informação necessária para localizar os dados

- O serviço de diretório também é responsável por manter armazenados os atributos relacionados a um arquivo:
 - **Entrada do Diretório:** Diretório consiste de uma lista de entradas com tamanho fixo (uma para cada arquivo) contendo um nome de arquivo (tamanho fixo), uma estrutura de atributos de arquivos, e um ou mais endereços de disco; MS/DOS e Windows;
 - **I-node:** nesse caso, o diretório de entrada é menor, armazenando somente o nome de arquivo e o número do i-node que contém os atributos; UNIX

- Tratamento de nomes de arquivos:
 - Maneira mais simples: limite de 255 caracteres reservados para cada nome de arquivo.
 - Maneira mais eficiente: tamanho do nome do arquivo é variável;

- Busca em diretório
 - **Linear**: lenta para diretórios muito grandes;
 - **Uma tabela Hash para cada diretório**: busca rápida, mas gerenciamento complexo;
 - **Cache de busca**: ótima para poucas consultas de arquivos;

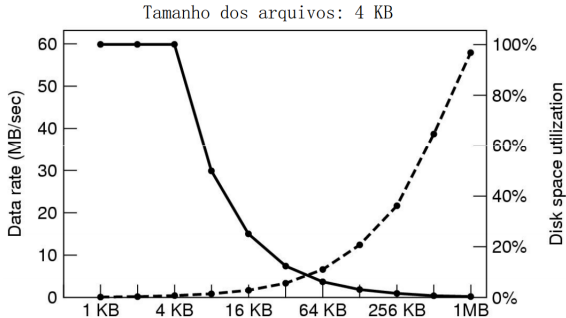
Implementando o Sistema de arquivos - Gerenciamento de espaço em disco

- Duas estratégias são possíveis para armazenar um arquivo de n bytes:
 - São alocados ao arquivo n bytes consecutivos do espaço disponível em disco;
 - Arquivo é espalhado por um número de blocos não necessariamente contínuos (blocos com tamanho fixo): A maioria dos sistemas de arquivos utilizam essa estratégia;

Implementando o Sistema de arquivos - Gerenciamento de espaço em disco

- Questão importante: Qual é o tamanho ideal para um bloco?
 - Se for muito grande, ocorre desperdício de espaço (fragmentação interna);
 - Se for muito pequeno, um arquivo irá ocupar muitos blocos, tornando o acesso/busca lento (fragmentação externa);
- Assim, o tamanho do bloco tem uma grande influência na eficiência de utilização do espaço em disco e no tempo de acesso(desempenho);

Implementando o Sistema de arquivos - Gerenciamento de espaço em disco



- Curva tracejada - Taxa de Dados x Tamanho do Bloco
- Curva contínua - Utilização do disco x Tamanho do Bloco

Implementando o Sistema de arquivos - Gerenciamento de espaço em disco

- Conflito entre performance (desempenho) e utilização do disco
→ blocos pequenos contribuem para um baixo desempenho, mas são bons para o gerenciamento de espaço em disco;
- UNIX: 1Kb;
- MS-DOS: 512 bytes a 32 Kb (potências de 2), depende do tamanho do disco;
- WINXP: 4Kb;
- Linux: 1Kb, 2Kb , 4Kb;

Implementando o Sistema de arquivos - Gerenciamento de espaço em disco

- Como controlar os blocos livres?
 - Lista ligada de blocos livres
 - Mapa de bits (bitmap)

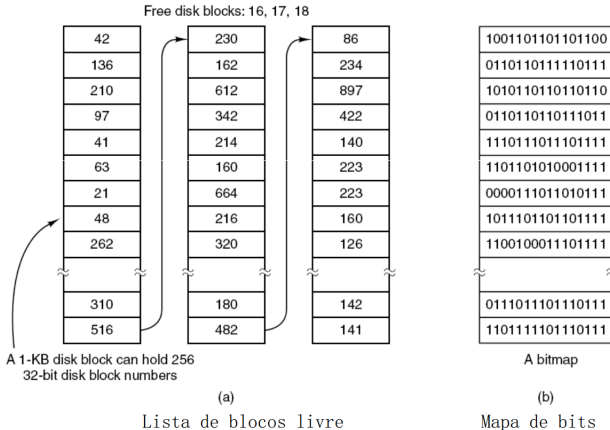
Implementando o Sistema de arquivos - Gerenciamento de espaço em disco

- **Lista ligada de blocos livres:** 32 bits para endereçar cada bloco, mantida no disco;
- Somente um bloco de ponteiros é mantido na memória principal → quando bloco está completo, esse bloco é escrito no disco;
- Vantagens
 - Requer menos espaço se existem poucos blocos livres (disco quase cheio);
 - Armazena apenas um bloco de ponteiros na memória;
- Desvantagens
 - Requer mais espaço se existem muitos blocos livres (disco quase vazio);
 - Dificulta alocação contínua;
 - Não ordenação;

Implementando o Sistema de arquivos - Gerenciamento de espaço em disco

- **Mapa de bits (bitmap):** depende do tamanho do disco
- Um disco com n blocos, possui um mapa de bits com n bits, sendo um bit para cada bloco;
- Mapa é mantido na memória principal;
- Vantagens
 - Requer menos espaço;
 - Facilita alocação contínua;
- Desvantagens
 - Torna-se lento quando o disco está quase cheio;

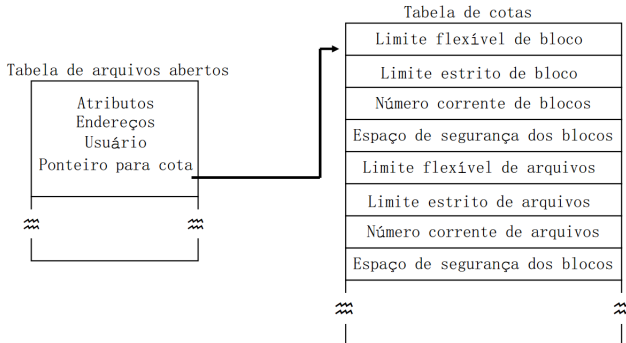
Implementando o Sistema de arquivos - Gerenciamento de espaço em disco



Implementando o Sistema de arquivos - Gerenciamento de espaço em disco

- Controle de cotas do disco: feito para que um usuário não ocupe muito espaço do disco → o administrador do sistema atribui para cada usuário uma cota máxima de espaço;
- Na memória principal: Tabela de arquivos abertos com ponteiro para uma tabela que mantém o registro de todas as cotas do usuário;

Implementando o Sistema de arquivos - Gerenciamento de espaço em disco



Implementando o Sistema de arquivos

- Algumas características importantes: Confiabilidade e Desempenho!!
- Confiabilidade: Backups e Consistência.
- Desempenho: Caching.

- Danos causados ao sistema de arquivos podem ser desastrosos;
- Restaurar informações pode ser (e geralmente é) custoso, difícil e, em muitos casos, impossível;
- Sistemas de arquivos são projetados para proteger as informações de danos lógicos, e não físicos;

- Backups: Cópia de um arquivo ou conjunto de arquivos mantidos por questão de segurança;
- Por que fazer backups?

Implementando o Sistema de arquivos - Confiabilidade

- Consistência: dados no sistema de arquivos devem estar consistentes;
- Crítico: blocos de i-nodes, blocos de diretórios ou blocos contendo a lista de blocos livres/mapa de bits de blocos livres;
- Diferentes sistemas possuem diferentes programas utilitários para lidar com inconsistências: fsck (UNIX) e scandisk (Windows).

Exemplos reais de sistemas de arquivos

- FAT (diversas versões) e NTFS;
- I-node do Unix e NFS;
- Extended File System (EXT2, EXT3, EXT4), criado em 1992 como uma evolução do Unix File System (UFS);
- ReiserFS;

- FAT surgiu por volta de 1976, sendo utilizada no SO do Intel 8086;
- FAT no MS-DOS:
 - Limite de nome de arquivo: 8 + 3(8.3) caracteres;
 - Hierarquia de diretórios: árvore começa no diretório raiz (root directory);
 - Não existe o conceito de diferentes usuários, portanto, todos os arquivos podem ser acessados por todos os usuários;
 - Cada entrada de diretório possui um tamanho fixo de 32 bytes;

- Três versões: FAT12, FAT16, FAT32 (VFAT), dependendo de quantos bits são utilizados para armazenar cada posição na FAT;
- FAT12:
 - Tamanho de bloco: 512bytes, 1K, 2K e 4K;
 - Tamanho da partição: Até 16Mb.
- FAT16
 - Tamanho de bloco: 8k, 16k e 32k;
 - Tamanho da partição: Até 2Gb.

- FAT32: iniciou-se com a segunda versão do Windows95.
 - Tamanho de bloco: 4k, 8k, 16k e 32k;
 - Tamanho da partição: Até 2Tb.
- Controle de blocos livres é feito utilizando a FAT;
- Windows98 utiliza FAT32 e permite arquivos com nomes longos (mais de 8 caracteres);

- NTFS não está baseado no Sistema de Arquivos FAT, mas se utiliza de algumas características do HPFS (High Performance File System - sistema de arquivos do OS/2);
- **Confiabilidade:** capacidade de se recuperar de problemas sem perda de dados; melhorara a tolerância a falhas;
- Segurança e Controle de Acesso (DAC - Discretionary Access Control): estabelece diretivas que permitem implementar controle de acesso em arquivos e diretórios, inexistente no Sistema de Arquivos FAT;
- Permite maiores partições no disco;
- Sistema de caracteres: UNICODE;
- Caminho: até 32.767 caracteres;

- Suporta Case Sensitive, no entanto, essa característica é perdida devido à Win32 API;
- Suporte à rede;
- Compressão de arquivos;
- Tamanho de blocos: 512bytes até 64Kb;
- Baseado no conceito de transações: tarefa é cumprida até o fim ou é abortada;
- Suporte à criptografia de arquivos;
- Links simbólicos.

- Cada arquivo é um conjunto de atributos, cada qual representado por um stream de bytes;
- Sistema de arquivos hierárquico: diretório de trabalho corrente, caminho relativo e absoluto;
- Master File Table (MFT): armazena a estrutura do NTFS e as informações sobre arquivos/diretórios; sequência linear de registros de 1Kb;
- Bloco de boot tem o endereço da MFT;

Sistemas de Arquivos

Características	FAT16	FAT32	NTFS
Sistemas	DOS, Win9x, NT	Win98, Win2000, WinXP, Win Vista	NT4, Win2000, WinXP, WinVista, 7, 8
Partição	2Gb	2Tb	Ilimitada
Número de arquivos	65000	Ilimitado	Ilimitado
Atributos	Conjunto básico	Conjunto básico	Vários
Segurança	Não	Não	Sim
Compressão	Não	Não	Sim

- Para o UNIX um arquivo é uma seqüência de 0s ou mais bytes contendo dados;
- Nenhuma distinção é feita entre arquivos ASCII, binários ou outros;
- Usa o conceito de i-nodes (64 bytes) associados aos arquivos: tabela.

- Diversos são os sistemas de arquivos utilizados pelo LINUX:
 - Ext2FS, Ext3FS, Ext4FS, Xia;
 - CFS, TCFS, VFS, GFV, NFS, HPFS, SYSV;
 - ReiserFS; JFS (IBM);
- Primeiro foi baseado no Minix;
- Como funcionam: Ext2FS, Ext3FS e ReiserFS??

Exercícios e Prova