

Gerência de Espaço Livre

Marcelo Johann

Lembrando: diretórios

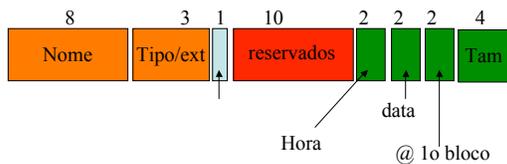
- Estrutura de dados para organizar os arquivos
 - Criação, remoção, listagem, *link*.
 - Árvore / DAG / Grafo (ciclos!)
- Implementados como arquivos especiais que contém uma seqüência de entradas
 - Tabela de entradas
 - Lista seqüencial, tabela de Hash, Árvore B...
 - Cada entrada descreve um arquivo/diretório contido no diretório
 - Exemplos: MS-DOS, NTFS, inode/Unix

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

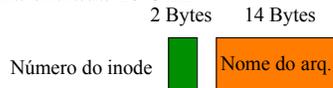
Aula 25 : Slide 2

Lembrando: dois exemplos

Uma entrada de diretório no MS-DOS



Uma entrada no Unix



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 3

DAG de diretórios & Links (atalhos)

- O *link* (atalho) possibilita o compartilhamento de um arquivo e/ou diretório.
 - Não há cópia física dos dados!
 - Possibilita a manutenção simples da coerência
 - Problema: pode criar laços!!
 - Problema para a procura.
 - Problema potencial na hora de deletar arquivos.
- Soft link:
 - Implementado através de um ponteiro sobre uma entrada física
 - Possibilita detectar ciclos, por isso lícito com diretórios
- Hard link
 - Implementado através de uma cópia física de uma entrada.
 - Impossibilita detecção de ciclos, proibido com diretórios.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 4

Qual espaço livre gerenciar?

- O disco é organizado em setores;
- O arquivo é organizado em registros (NTFS)...
 - O registro contém (endereços de) blocos.
 - Há uma tradução registros / blocos
- Blocos são um múltiplo de setores
 - Escolha nem sempre fácil!
- Um bloco pode ter um tamanho fixo ou variável.
- Gerenciamento de espaço livre envolve 2 problemas:
 - Decidir o tamanho de um bloco;
 - Administrar/recuperar os blocos livres/ocupados no disco.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 5

Influência do tamanho do bloco

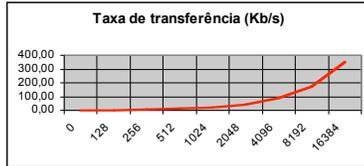
- No mínimo, 1 bloco = 1 setor
 - Se for menor, perde-se (muito) tempo ao ler os dados no disco.
- No máximo 1 cilindro
 - Desperdiço de espaço!
 - Cf. [Mullender & Tannenbaum, 1984]: tamanho médio de arquivos no Unix é 1 KB.
- Entre os dois:
 - deve-se escolher um tamanho que não perca muito tempo em seek/atraso rotacional, nem muito espaço.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 6

Exemplo de meio-termo (1)

- Seja um HD com 32 KBytes/trilha, 16,67 msec de tempo de rotação e seek médio de 30 msec.
- **Tempo de transferência** de um bloco de k Bytes (msec) = $T(k) = 30 + 8,33 + (k / 32768) * 16,67$
- Taxa de transferência = $k / T(k)$

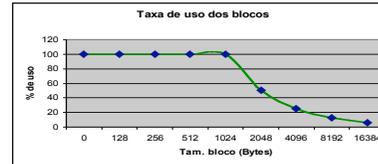


INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 7

Exemplo de meio-termo (2)

- E a taxa de uso do HD?
- Supondo que só se tenha arquivos de 1 KB:
 - Com blocos $k < 1$ KB, usa-se 100% dos blocos;
 - Com blocos $k > 1$ KB, usa-se $1 \text{ KB} / k \text{ Bytes}$, $(1000/k) * 100 \%$ dos blocos.
- $k = 1 \text{ KB} \Rightarrow 100\%$ de uso ; $k = 8 \text{ KB} \Rightarrow 1/8 = 12 \%$



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 8

Exemplo de meio-termo (3)

- Com essa configuração, um meio-termo será $k = 1 \text{ KB}$, 2 KB...
- Altamente dependendo:
 - Das características do HD
 - Da distribuição do tamanho dos arquivos...
 - Por exemplo: o surgimento dos vídeos / arquivos de som alterou recentemente o tamanho médio dos arquivos em um HD!
- O tamanho do bloco deve acompanhar o desenvolvimento dos HD e dos Sis. Op....

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 9

Vetor de bits (bitmap)

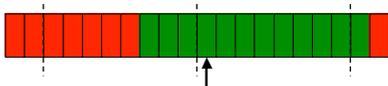
- Cada bloco é representado por 1 bit
 - 1 se estiver livre, 0 se estiver usado.
- Implementação eficiente
 - Instrução de HW específica para retornar o 1º bit valendo 1.
- Problema: utiliza tantos bits quanto o número de blocos
 - Bitmap armazenado na RAM!
 - HD de 80 GB, blocos de 1 KB $\Rightarrow 80 \text{ Mbits} = 10 \text{ MBytes}$...
 - Obs: com partições/clusters, reduz-se o bitmap.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 10

Alocação agrupada com bitmap (ext2fs)

- Dois problemas com a escolha do 1o bit livre:
 - Achar rapidamente um bit 1
 - Procura byte por byte!
 - Essa solução "estimula" a fragmentação
- Solução:
 - Procura-se um byte livre;
 - Vai-se para trás até o último bit ocupado.
 - Pre-alocação dos 8 blocos pela frente

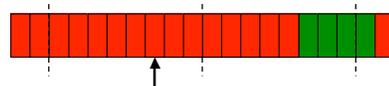


INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 11

Alocação agrupada com bitmap (ext2fs)

- Dois problemas com a escolha do 1o bit livre:
 - Achar rapidamente um bit 1
 - Procura byte por byte!
 - Essa solução "estimula" a fragmentação
- Solução:
 - Procura-se um byte livre;
 - Vai-se para trás até o último bit ocupado.
 - Pre-alocação dos 8 blocos pela frente



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 12

Interesse deste esquema

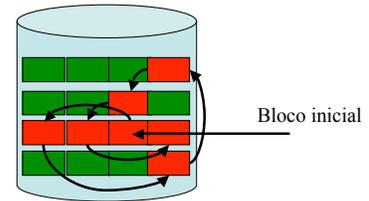
- A procura Byte por Byte é rápida.
 - Mais do que se fosse bit por bit.
- A procura para trás minimiza a fragmentação
- A pre-alocação
 - Reduz a fragmentação quando se escreve em vários arquivos intercalados
 - Agiliza a alocação dos blocos.
- Blocos pre-alocados que não forem usados são devolvidos quando se fecha o arquivo.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 13

Lista encadeada

- Mantém-se um ponteiro (endereço) do bloco inicial na lista de endereços de blocos disponíveis;
- Os blocos são encadeados numa lista
 - Cada elemento da lista é um bloco
- Obs: mantém numa *cache* parte da lista
 - E.g. o 1o bloco, frequentemente suficiente.



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 14

Blocos encadeados (cont.)

- A lâmina anterior esconde um "detalhe": na lista, cada bloco contém (mais de um) endereços de blocos livres...
- Um bloco de 1 KB pode conter 256 endereços (de 4 Bytes) de blocos livres
 - 1 dos 256 é o ponteiro para o próximo bloco de endereços de blocos...
 - 1 HD de 80 GB contém 80 Mblocos e é preciso de $80 M / 255 = 329\ 000$ blocos para gerenciar os blocos livres.
- Comparação:
 - Bitmap requer menos espaço
 - Lista implica em armazenar na Ram apenas 1 bloco por vez (1 KB)

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

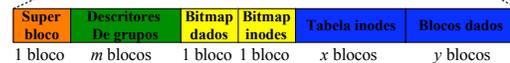
Aula 25 : Slide 15

Exemplo: ext2fs

- Sistema de arquivos "clássico" do Linux
 - Inspirado pelo UNIX BSD (FFS)
- Decompõe uma partição ext2fs em **grupos**
 - Dessa forma, agrupa blocos.



- Cada grupo tem a estrutura seguinte:



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 16

Descrição dos grupos no ext2fs

- Super-bloco:
 - Contém dados relevantes a toda a partição (duplicado em cada grupo)
 - Número mágico da partição,
 - número de *mounts*,
 - tamanho do bloco,
 - tamanho do grupo (*m*),
 - ponteiro para o 1o inode do sistema de arquivos (*i*),
 - Número de inodes, de inodes livres, de blocos livres...
- Descritor do grupo:
 - 1 entrada por grupo
 - Cada entrada fornece:
 - o endereço do bloco onde está o bitmap dos blocos livres;
 - o endereço do bloco onde está o 1o inode na tabela.
 - o endereço do bloco do bitmap dos inodes
 - o endereço do bloco do bitmap dos blocos

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 17

Descrição do grupo no ext2fs (cont.)

- O ext2fs usa bitmaps para gerenciar os blocos e inodes livres;
 - Cópias parciais através de caches!
- A tabela de inodes contém os endereços das cópias na memória dos inodes disponíveis (livres ou ocupados) no grupo;
 - O bitmap indica quais usar num dado momento.
- Por fim, a tabela dos blocos aponta para os blocos disponíveis.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 18

Sistema de arquivos NTFS (Windows 2000+)

- **Objetivos principais:**
 - Confiabilidade e tolerância a falhas.
 - Garante a manutenção dos metadados.
 - logging
 - Eficiência
 - Links, compressão, arquivos esparsos...
- **Organiza os discos em volumes (partição lógica);**
- **O cluster (bloco) varia ao formatar o volume**
 - de 1 até 2^o setores do disco
 - 512 Bytes, ... Alguns Kbytes.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 19

NTFS, MFT e metadados

- **MFT = Master File Table**
 - É um arquivo especial
 - Tabela de registros de 1 KB
 - 1 registro por arquivo no volume
 - Inclui 1 registro para o arquivo que implementa a própria MFT.
- **Além da MFT, NTFS define uma série de arquivos de metadados.**
 - Exemplo: meta-arquivo de log.
- **O primeiro registro da MFT contém o setor de boot, usado para carregar o sistema de arquivos na memória.**
 - Espelhado no meio da MFT, caso haja um problema.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 20

Registros da MFT

- **Registros de metadados:**
 - Diretório raiz (I)
 - Este registro contém a lista dos arquivos / diretórios encontrados na raiz;
 - Alocação dos blocos do volume
 - Bitmap
 - Blocos danificados
 - ...
- **Registro de arquivo:**
 - Sequência de pares (nome atributo, valor)
 - Alguns atributos: nome arquivo, informações de acesso, dados...
 - Registro residente: para pequenos arquivos, todos os atributos cabem no registro da MFT (1 KB)
 - Fala-se de atributos residentes
 - Registro estendido: blocos fora da MFT são usados para armazenar parte dos "grandes" atributos (e.g. dados).

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 21

Os registros da MFT

0	MFT	} Reservados aos Meta-dados predefinidos
1	Espelho MFT	
2	Arquivo de log	
3	Arquivo do volume	
4	Tabela de def. atributos	
5	Raiz	
6	Bitmap de clusters livres	
7	Setor de boot	
8	Setores estragados	
9	Clusters estragados	
10	Segurança	
11	Dir. meta-dados estendidos	
12	Não usado...	
16	Arquivo/diretório usuário	

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 22

Diretório no NTFS

- **Um diretório é um registro especial**
 - Ou seja é um arquivo especial
- **Inclui um atributo "diretório"**
 - Nome
 - Atributos normais (direitos de acesso, tamanho...)
 - Ponteiros para os registros da MFT que implementam seus arquivos ou sub-diretórios.
 - Podem ser atributos residentes
 - Podem ser não-residentes.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 23

Continuando...

- **Confiabilidade e desempenho**

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 25 : Slide 24