

Implementação de Diretórios

Marcelo Johann

Lembrando: implementação de arquivos & sistemas de arquivos...

- Arquivos são constituídos de seqüências de blocos
 - Bloco mapeado em setores físicos do disco;
 - Alocação contígua de blocos :o(
 - Fragmentação vs. Bom desempenho
 - Uso de listas
 - Listas com índice
 - FAT
 - Inode (Unix/Linux)
- Sistemas de arquivos incluem meta-dados
 - Setor de boot
 - Endereço do diretório raiz
 - Cópia espelhada da FAT (no caso do DOS/NTFS).

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

Aula 24 : Slide 2

Plano

- Implementação de diretórios:
 - Do ponto de vista do usuário:
 - Organização dos diretórios;
 - Links e ciclos na organização.
 - Como implementar a estrutura de dados para gerenciar diretórios?
 - Estabelecer um relacionamento entre nome e blocos, entre meta-dados e nome.
 - Listas, tabelas de hash, B-trees...
 - Casos de estudo:
 - MS-DOS, Unix, NTFS.

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

Aula 24 : Slide 3

Diretório & Arquivos

- O arquivo é a unidade básica de dados (do ponto de vista do usuário)
- O diretório é a unidade básica para organizar os arquivos.
 - Em geral, implementado como sendo um arquivo "especial".
 - Chamado "pasta" em alguns sistemas...
- O diretório deve associar, para cada arquivo/diretório nele contido:
 - Um nome definido pelo usuário;
 - Informações de contabilidade;
 - Um ponteiro sobre o arquivo.

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

Aula 24 : Slide 4

Operações com diretórios

- **Criação**
 - Deve-se acrescentar uma entrada na estrutura de dados "diretório" para incluir o novo diretório/arquivo.
 - Possivelmente, deve-se re-organizar as entradas.
- **Remoção**
 - Deve-se tirar uma entrada;
 - Obs: problema potencial ao remover um diretório inteiro! Fazer o que se não está vazio?
 - Possivelmente, deve-se re-organizar as entradas.
- **Acesso/listagem**
 - Deve retornar a lista das entradas;
 - Para tanto, precisa ser "aberto".

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

Aula 24 : Slide 5

Diretório de um nível

- Todos os arquivos estão em **um diretório** só, no sistema todo.
- Ausência total de controle de acesso...
 - Por exemplo, arquivos executáveis e/ou de sistema estão misturado com os dados dos usuários.
- Falta de eficiência para procurar os arquivos.
- Conflito entre os nomes dos arquivos
 - Todos os usuários devem evitar usar nomes parecidos!
- Usado em sistemas de arquivos básicos & antigos
 - Floppies.

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

Aula 24 : Slide 6

Diretório de 2 níveis

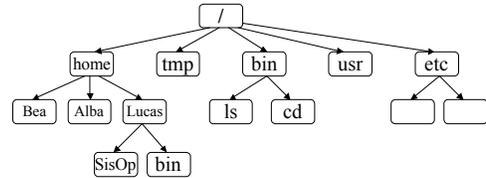
- Um primeiro nível distingue (por exemplo) contas de usuários;
 - Serve também para isolar diretórios do sistema.
- Cada usuário tem todos seus arquivos em um único diretório, pessoal.
- Pode-se compartilhar arquivos
 - É preciso prefixar o nome do arquivo pelo diretório onde se encontra.
- Pode-se compartilhar um diretório inteiro
 - Bin/ com os executáveis do sistema.

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

Aula 24 : Slide 7

Árvore de diretórios

- Generalização do mecanismo com 2 níveis.



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

Aula 24 : Slide 8

Caminho (*path*)

- O caminho é a seqüência de diretórios a serem percorridos até chegar a um arquivo.
- Caminho **absoluto**: a partir da raiz
 - Começa com '/' (ou '\', em MS-DOS/Windows)
 - /home/nmaillard/bin/gcc
- Caminho **relativo**:
 - Diretório corrente (".") = diretório onde o usuário está trabalhando quando dispara o processo.
 - Deve ser incluído na struct proc!
 - O caminho relativo começa a partir do diretório corrente
 - bin/gcc
 - ../bea/bin/gcc
- Os diretórios '.' e '..' designam o diretório corrente e pai.
- Existe uma variável de ambiente **PATH** que indica os caminhos default onde procurar um arquivo executável.
 - Um processo mantém uma variável para seu diretório corrente

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

Aula 24 : Slide 9

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 24 : Slide 10

Implementação de diretório

- O diretório provê o acesso aos arquivos e outros diretórios nele contidos.
 - Deve mapear o nome simbólico com a estrutura de dados que implementa o arquivo
 - Arquivo todo, endereço do 1o bloco, inode...
 - Também pode haver mapeamento para um outro diretório.
- O diretório será implementado como uma **seqüência de entradas**, cada qual descrevendo um dos diretórios/arquivos nele contido.
 - Várias soluções para implementar a seqüência

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

Aula 24 : Slide 11

Possibilidades de implementação

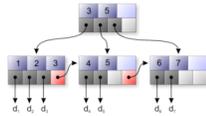
- Lista linear de entradas
 - Pesquisa linear... :o(
 - Cada inserção necessita uma procura para evitar nomes repetidos!
 - Simples :o)
 - Uma forma de otimizar ordenar a lista
 - Sobrecusto à inserção
- Tabela de Hash
 - Lista linear + índice
 - Deve-se gerenciar as colisões!

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

Aula 24 : Slide 12

Implementação com Árvores B (B-trees)

- Estrutura de dados para agilizar as procuras.
 - Altura $\log(n)$, com n entradas.
 - Usado para dicionários
- Baseado em chaves, usadas para definir:
 - O número de filhos de cada nó
 - Um filho / intervalo de valores da chave
 - Todas as folhas estão na mesma altura
- Pesquisa parecida à árvore binária
 - Compara-se as chaves para orientar a procura.
- Inserção de chave mais complicada!
 - Divide-se um nó folha, e migra-se a chave para cima.

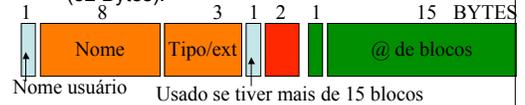


INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 24 : Slide 13

Exemplo de implementação de entrada

- Implementa poucas informações
 - Nome + tipo (extensão): foo.txt
 - Identificador do usuário dono do diretório/arquivo
- Exemplo simples: CP/M (1986)
 - 1 diretório único
 - Endereços dos Blocos dos arquivos armazenados também na estrutura do diretório
 - Uma entrada do diretório é da forma seguinte (32 Bytes):

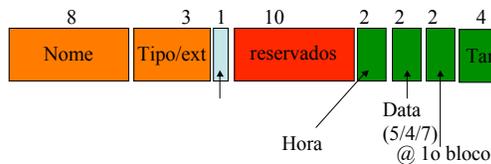


INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 24 : Slide 14

Implementação no MS-DOS

- Árvore de diretórios!
- Cada entrada tem comprimento 32 Bytes
 - Nome, extensão...
 - Ponteiro para o primeiro bloco na FAT.
 - Os outros blocos do arquivo se encontram graças à FAT.
 - Se o bloco apontado na FAT contiver a mesma estrutura, aninham-se os diretórios.

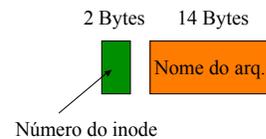


INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 24 : Slide 15

Implementação no Unix

- Muito simples, devido ao uso de inodes.
 - As informações sobre o arquivo/diretório estão armazenadas no inode (atributos):
 - Nome, extensão, data/hora de criação...



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

Aula 24 : Slide 16

Procura por um arquivo

- Procura-se o arquivo /alguma/coisa/foo.txt
- Localiza a raiz / (vide entrada no sistema de arquivos!)
- É um diretório (inode, por exemplo).
 - Vai ser consultado para achar o que tem dentro.
 - Procura-se o diretório 'alguma'
 - Vai achar o inode correspondendo
 - Vai carregar o 1o bloco do disco, que implementa o diretório.
- É um diretório... Se repete o procedimento
 - Acha o inode associado a 'coisa'
 - Carrega-o do disco
- Afinal, chega-se a procurar 'foo.txt'
 - Obtém-se o inode apropriado... E os blocos de dados.
- Obs: pode ter erro se algum nome está errado!

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

Aula 24 : Slide 17

Próxima aula...

- Gerenciamento de espaço livre.**

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

Aula 24 : Slide 18