
MAC5753 - Sistemas Operacionais

Daniel Macêdo Batista

IME - USP, 15 de Outubro de 2020

Introdução aos sistemas
de arquivos

Arquivos

Introdução aos sistemas de arquivos

Arquivos

Introdução aos sistemas de arquivos

Dispositivos de armazenamento

Introdução aos sistemas
de arquivos

Arquivos

- Necessários porque o espaço virtual (físico) da memória não é suficiente para os processos salvarem dados
- Necessários porque os processos (programas) precisam estar salvos em algum lugar, principalmente falando de computadores de propósito geral
- Necessários porque usuários terão necessidade de salvar seus dados
- Importante manterem os dados mesmo quando o computador for desligado
- Importante que os dados sejam independentes dos processos (Um mesmo dado pode ser útil para vários processos em instantes de tempo diferentes)

Dispositivos de armazenamento

Introdução aos sistemas
de arquivos

Arquivos

- Discos magnéticos (hoje em dia discos de estado sólido – SSDs)
- Discos flexíveis, CDs, DVDs (hoje em dia pendrives)

Dispositivos de armazenamento

Introdução aos sistemas
de arquivos

Arquivos

- Do ponto de vista do processo devem possibilitar:
 - Ler o bloco k
 - Escrever no bloco k

Dispositivos de armazenamento

Introdução aos sistemas
de arquivos

Arquivos

- Para possibilitar as ações de leitura e escrita de blocos é necessário responder essas 3 perguntas:
 - Como encontrar uma informação?
 - Como impedir que um usuário acesse informações de outros?
 - Como descobrir quais blocos estão livres?

Dispositivos de armazenamento

Introdução aos sistemas
de arquivos

Arquivos

- Para possibilitar um acesso padronizado das aplicações é necessário abstrair a forma como os dados são armazenados (de forma similar ao que foi feito com o acesso ao processador – processo – e como será feito com a memória física – memória virtual)
- A abstração vai ser feita com **arquivos**

Arquivos

Introdução aos sistemas
de arquivos

Arquivos

- Unidades lógicas de informação criadas por processos
- Um dispositivo de armazenamento contém vários arquivos
- Informações armazenadas nos arquivos são persistentes
- Processos podem ler e escrever em arquivos (respeitando as permissões)

Sistema de arquivos

Introdução aos sistemas de arquivos

Arquivos

- Os arquivos são gerenciados pelo SO por uma parte dele chamada de **sistema de arquivos**
- Existem vários tipos de sistemas de arquivos que definem diferentes respostas para as 3 perguntas mostradas anteriormente (busca de informação, segurança e gerenciamento de espaço livre)

Sistema de arquivos

Introdução aos sistemas
de arquivos

Arquivos

- Do ponto de vista do usuário o mais importante é saber como o sistema de arquivos se organiza em nomes
- Vamos considerar os sistemas de arquivos de SOs Unix-like com arquivos separados por '/', sendo a '/' inicial o diretório de mais alta hierarquia (diretório raiz) e os demais níveis sendo delimitados por outras '/'

Introdução aos sistemas
de arquivos

▷ Arquivos

Arquivos

Nome de arquivos

Introdução aos sistemas
de arquivos

Arquivos

- Lembrando: arquivos são abstração que permitem armazenar informação num disco (vamos usar o termo 'disco' mesmo se o dispositivo não for um disco) e lê-la depois
- Como fazer para abstrair todas as informações de baixo nível? Usando nomes (o ser humano lembra de nomes mais fácil do que de números – Na Internet isso também é considerado)
- Quando um processo cria um arquivo, o arquivo precisa de um nome e esse nome vai ser usado para o arquivo ser recuperado no futuro

Regras para nomes de arquivos

Introdução aos sistemas
de arquivos

Arquivos

- As regras variam de sistema para sistema
- A maioria dos SOs permitem que os arquivos sejam nomeados com strings de 1 a 8 letras como nomes de arquivos. Dígitos e números também costumam ser permitidos
- Como há a interação dos nomes dos arquivos no shell, caracteres que podem afetar a interação no shell devem ser evitados
- Obviamente o separador de diretórios não pode ser usado nos nomes dos arquivos

Regras para nomes de arquivos

Introdução aos sistemas
de arquivos

Arquivos

- No Unix os nomes de arquivos são sensíveis ao contexto (home é diferente de Home)
- Muitos sistemas operacionais definem nomes de arquivos como tendo duas partes, separados por um ponto. Isso ajuda o usuário a identificar o que é o arquivo (.c, .ods, .html, etc...). Essa segunda parte é chamada de extensão
- Em alguns sistemas de arquivos a extensão inclusive é levada ao pé da letra e é considerada como definindo o tipo de arquivo (No Unix não é assim), o que pode limitar quais programas abrirão aquele arquivo
- Ainda no Unix, embora não haja regras a respeito das extensões é bom utilizá-las. Para ter certeza do tipo de arquivo o utilitário `file` pode ser usado

Estruturas de arquivos

Introdução aos sistemas
de arquivos

Arquivos

- Há 3 formas principais de estruturar um arquivo:
 - Sequência de bytes não estruturada
 - Sequência de registros de comprimento fixo
 - Árvore de registros

Sequência de bytes

Introdução aos sistemas
de arquivos

Arquivos

- Para o SO é uma sequência de bytes armazenados no disco. O SO nem sabe que corresponde a um arquivo. Qualquer significado além de bytes deve ser dado por programas do usuário
- No Unix é assim
- Essa é a estrutura com maior flexibilidade: programas podem armazenar qualquer quantidade de bytes com qualquer conteúdo. Útil para sistemas de propósito geral

Sequência de registros

Introdução aos sistemas
de arquivos

Arquivos

- O SO entende registros e nesse caso um arquivo é uma sequência de registros
- 1 registro = 1 bloco de dados de tamanho fixo e com uma certa estrutura interna (por exemplo um cabeçalho que defina o tipo de informação armazenada no registro)
- Útil em sistemas de propósito específico em que sempre será necessário ler ou escrever múltiplos de 1 registro (nesse caso não faz sentido pensar em uma unidade menor que 1 registro)
- Na época dos mainframes baseados em cartão perfurado fazia sentido ter 1 registro com o tamanho de 80 caracteres pois cada cartão perfurado tinha 80 colunas
- Ainda na época dos primeiros mainframes, registros de 132 caracteres também existiam pois as impressoras conseguiram imprimir linhas com 132 colunas

Árvore de registros

Introdução aos sistemas
de arquivos

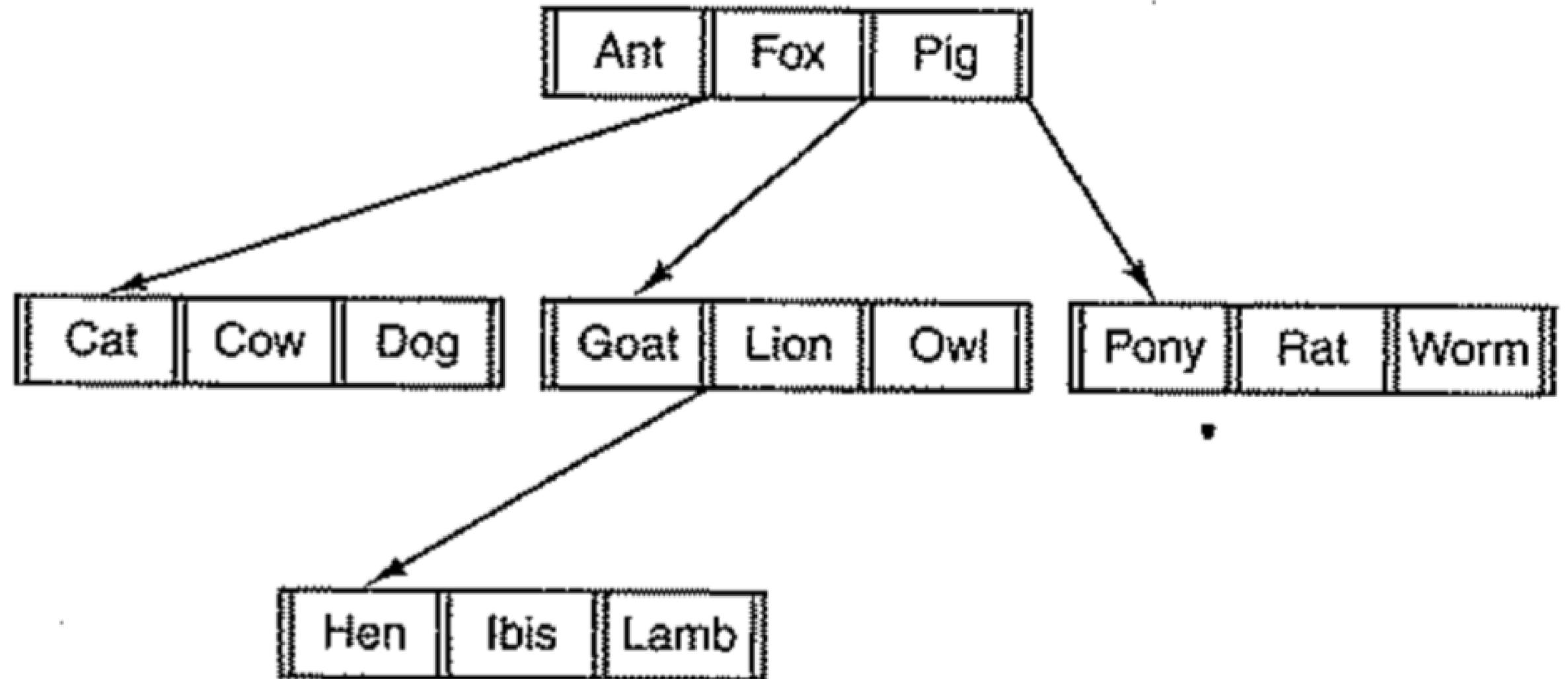
Arquivos

- O SO entende que um arquivo vai ser um conjunto de registros organizados em árvores
- Os registros nesse caso de árvores não precisam ter os mesmos tamanhos
- Cada registro tem um campo **chave** em uma posição fixa no registro (cabeçalho por exemplo). A árvore pode ser ordenada pela chave para permitir uma busca rápida dentro do arquivo
- Útil para operações de busca dentro do arquivo (não sabe onde dentro do arquivo está a informação mas quer chegar rápido nela)
- Nesse caso o SO tem que entender a organização do arquivo para saber como colocar dados lá dentro
- Faz sentido em mainframes orientados a armazenamento de dados (não muito processamento)

Árvore de registros

Introdução aos sistemas
de arquivos

Arquivos



Tipos de arquivos

Introdução aos sistemas
de arquivos

Arquivos

- Na maioria dos sistemas há pelo menos dois tipos de arquivos:
 - Arquivos regulares
 - Diretórios
- Arquivos regulares contêm informações do usuário
- Diretórios são usados para manter a estrutura do arquivo (pode ter outros diretórios ou arquivos regulares “em baixo”)

Tipos de arquivos

Introdução aos sistemas
de arquivos

Arquivos

- No caso do Unix há ainda:
 - Arquivos especiais de caracteres
 - Arquivos especiais de bloco
 - Links
- Arquivos especiais de caracteres são usados para E/S de dados em formato de caracter (terminais e impressoras por exemplo)
 - /dev/tty0
- Arquivos especiais de bloco são usados para modelar discos –
 - /dev/sda
- Links são usados para servir como ponteiros para os arquivos (ou diretórios) reais – manipulados com o comando `ln`. Os links podem ser *symbolic* ou *hard*

Arquivos regulares

Introdução aos sistemas
de arquivos

Arquivos

- Texto puro: possui linhas de texto terminadas por algum caracter de fim de linha (no Unix é o CR – *Carriage Return*, que em hexadecimal é 0A). A codificação desses arquivos costuma ser ISO-8859-1 ou UTF-8 (UTF-8 suporta mais caracteres)
- Vantagem de arquivos texto é que são universalmente compatíveis. Desvantagem é que podem ocupar muito espaço em casos onde não é necessário armazenar informações em texto (por exemplo, se quiser guardar números)
- Binário: possui bytes sem uma estrutura pré-estabelecida. A estrutura vai depender do tipo de programa que cria / abre esse arquivo

Arquivos binários

Introdução aos sistemas
de arquivos

Arquivos

- Um exemplo de arquivo binário é um executável, um arquivo que contém um programa que será carregado na memória
- Para o Unix executar um arquivo binário ele deve obedecer uma certa estrutura
- Esses arquivos costumam começar com um cabeçalho para já no início o SO saber se pode executar o arquivo
- No caso de um executável, precisa ter pelo menos: o código e os dados do programa, além do cabeçalho no início informando o tamanho desses pedaços de informação e começando com o *Magic Number*
- Para saber mais sobre os vários tipos de arquivos:
 - `man magic`
 - `view /usr/share/file/magic`
 - `xxd, hexdump`