

Curso de

Sistemas Operacionais

Gerenciamento de Arquivos Parte 1

Prof. Dr. Robson Augusto Siscoutto
e-mail: robson.siscoutto@unoeste.br

Sistemas Operacionais

Gerenciamento de Arquivos

- Introdução;
- Arquivos;
- Diretórios;
- Implementação do Sistema de Arquivos:
 - Implementação de Arquivos;
 - Implementação de Diretórios;
 - Arquivos Compartilhado;
 - Gerenciamento de Espaço em Disco
 - Monitoração dos Blocos Livres
 - Desempenho do Sistema de Arquivos
 - Sistema de arquivos LOG Estruturado
 - Exemplos de Sistemas de Arquivos

Gerenciamento de Arquivos

Introdução

- **Três requisitos fundamentais quanto ao problema do armazenamento de informações a longo prazo:**
 1. Deve ser possível o armazenamento de grande quantidade de informação;
 - O processo possui uma quantidade limitada de informação dentro de seu próprio espaço de endereçamento;
 - Contudo, a capacidade de memória está restrita ao tamanho do espaço de endereçamento virtual;
 - Para algumas aplicações é suficiente o espaço, mas para outras é pequena demais: reservas de passagens aéreas, bancos ou sistemas corporativos;

Gerenciamento de Arquivos

Introdução

2. A informação, dentro do espaço de endereçamento do processo, deve sobreviver ao término do processo que a estiver usando;

- Em algumas aplicações (p.ex. banco de dados), a informação precisa ficar retida por semanas, meses ou até anos;
 - É inaceitável que a informação em uso pelo processo desapareça quando ele é encerrado;
- Se uma falha no computador eliminar o processo, a informação não deve desaparecer;

Gerenciamento de Arquivos

Introdução

3. Vários processos devem ser capazes de manipular concorrentemente a informação (ao mesmo tempo);

-
- *Problema:* Informações dentro do espaço do processo são de propriedade dele, ou seja, nenhum outro processo tem acesso;
- *Solução:* Tornar a informação independente de qualquer processo;

Gerenciamento de Arquivos

Introdução

- **Três requisitos essenciais para o armazenamento da informação por longo prazo:**
 1. Deve ser possível armazenar uma quantidade muito grande de informação;
 2. A informação deve sobreviver ao término do processo que a usa;
 3. Múltiplos processos têm de ser capazes de acessar a informação concorrentemente;

Gerenciamento de Arquivos

Introdução

- **Solução usual para todos esses problemas:**
 - Armazenar a informação em discos e outros meios externos em unidades chamadas **arquivo**:
 - **Persistentes:**
 - A informação não pode ser afetada pela criação e pelo termino do processo;
 - O arquivo só desaparece quando seu proprietário o removê-lo;
 - **Gerenciados pelo Sistema Operacional:**
 - Estrutura, nomes, acessos, proteção, uso, etc 7

Gerenciamento de Arquivos

Arquivos

- **Nomeação de Arquivos:**

- As regras exatas para se dar nome a um arquivo variam de sistema para sistema;
 - Alguns sistemas permitem de 1 a 8 letras;
 - Outros permitem com até 255 letras;
 - Alguns sistemas diferenciam letras minúsculas de maiúsculas;
 - Unix;
 - Suporte de nomes em duas partes, separadas por ponto;
 - nome.extensão
 - MS-DOS

Gerenciamento de Arquivos

Arquivos

- **Nomeação de Arquivos:**

- Extensões por convenções:

- UNIX:

- Simplesmente para lembrar o proprietário do programa que deve ser utilizado;
 - Não oferece nenhum tipo de informação real ao Sistema Operacional

- Windows

- registra as extensões no S.O. indicando qual programa possui aquela extensão;
 - a partir da extensão executa o programa correto, tendo o arquivo como parâmetro.

Gerenciamento de Arquivos

Arquivos

- **Nomeação de Arquivos:**

Extension	Meaning
file.bak	Backup file
file.c	C source program
file.gif	Compuserve Graphical Interchange Format image
file.hlp	Help file
file.html	World Wide Web HyperText Markup Language document
file.jpg	Still picture encoded with the JPEG standard
file.mp3	Music encoded in MPEG layer 3 audio format
file.mpg	Movie encoded with the MPEG standard
file.o	Object file (compiler output, not yet linked)
file.pdf	Portable Document Format file
file.ps	PostScript file
file.tex	Input for the TEX formatting program
file.txt	General text file
file.zip	Compressed archive

Extensões típicas

Gerenciamento de Arquivos

Arquivos - Estrutura interna do Arquivo

- **Três tipos de Arquivo:**

- (a) Seqüência de Bytes

- Nenhuma estrutura lógica – Aplicação. Responsável;
 - tanto o Windows e UNIX implementam isso;
 - Maior flexibilidade – colocar qualquer coisa no arquivo, sem interferência do S.O.;

- (b) Seqüência de Registros de Tamanho Fixo

- Cada um alguma estrutura interna;
 - Objetivo: leitura e gravação de registros ao invés de bytes;

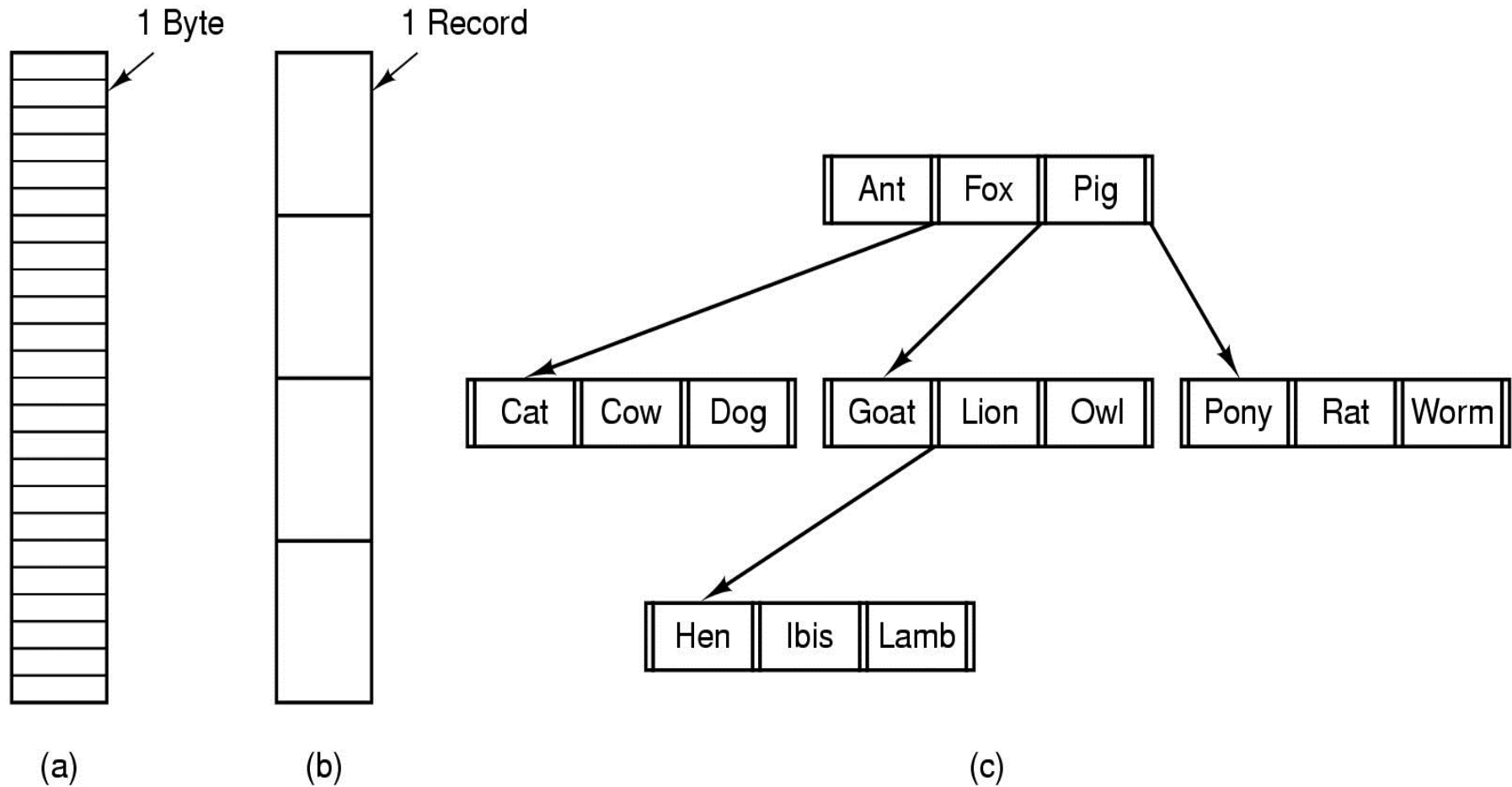
Gerenciamento de Arquivos

Arquivos - Estrutura interna do Arquivo

- **Três tipos de Arquivo:**
 - (c) Árvore de Registros
 - Ordenada por campo chave;
 - Nem todos os registros têm os mesmos tamanhos;
 - Amplamente utilizado em computadores de grande porte
 - Processamento de dados comerciais;

Gerenciamento de Arquivos

Arquivos - Estrutura interna do Arquivo



Gerenciamento de Arquivos

Arquivos

- **Tipos de Arquivos:**

- **Regulares:**

- São aqueles que contem informações do usuário;

- **Diretórios:**

- são arquivos do sistema que mantêm a estrutura do sistema de arquivo;

- **Arquivos especiais de caracteres (E/S);**

- são relacionados a entrada/saída e usados para modelar dispositivos de E/S, como terminais, impressoras e redes;

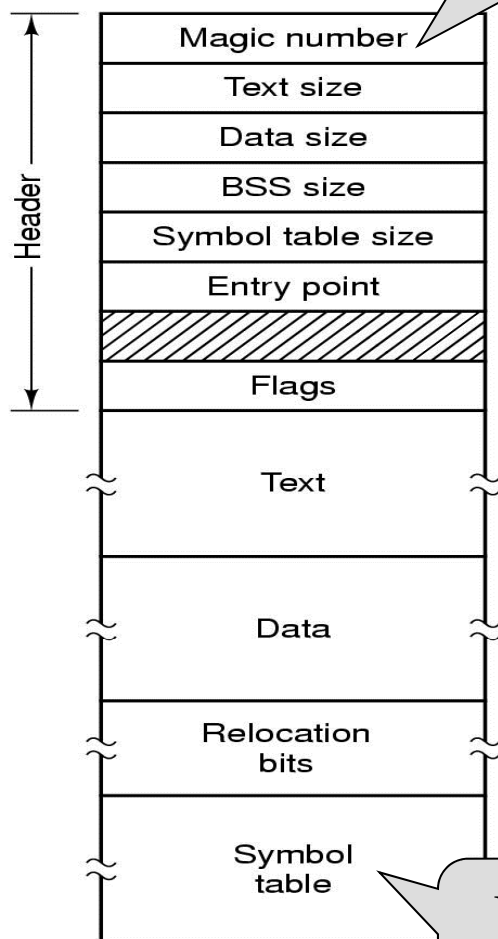
- **Arquivos especiais de Blocos (modelar discos)**

- são usados para modelas discos;

Gerenciamento de Arquivos

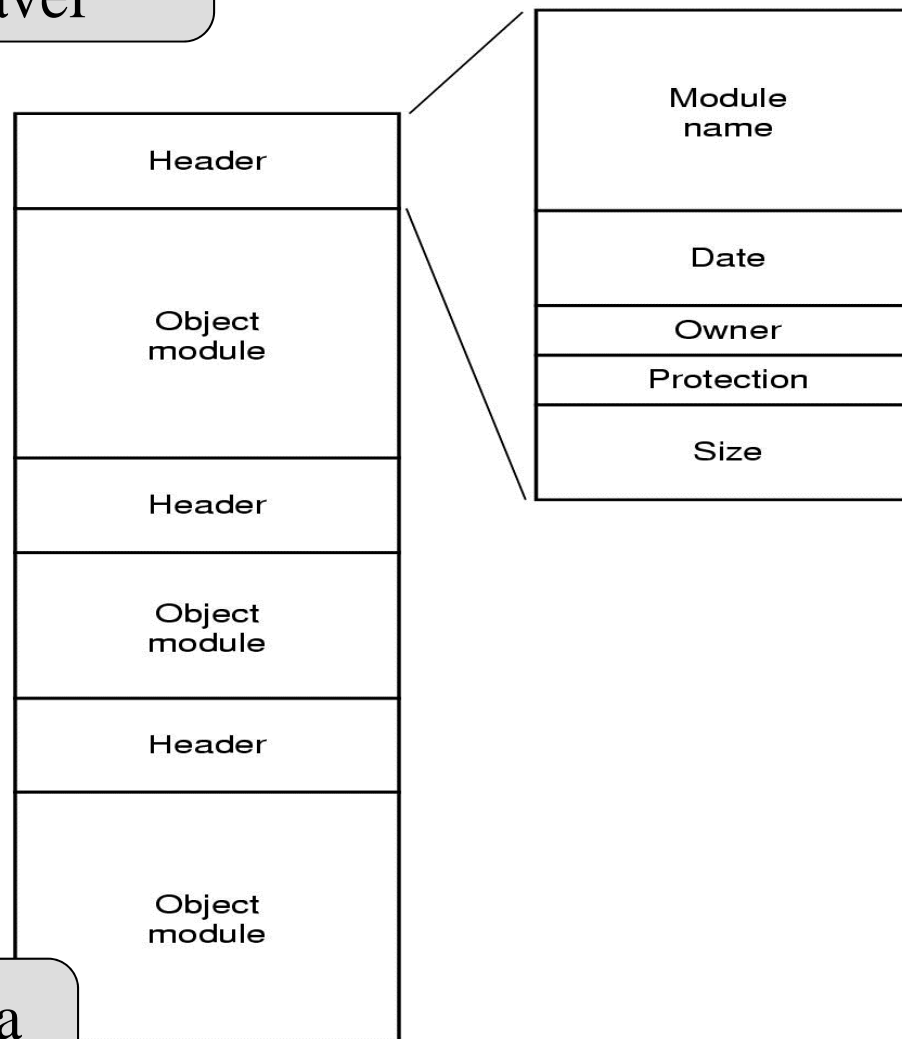
Arquivos

Identifica como executável



(a)

Usada na Depuração



(b)

Gerenciamento de Arquivos

Arquivos - Acesso Interno a Arquivos

- **Acesso Seqüencial:**

- Ler todos os Bytes/Registros partindo do início do arquivo;
- Não era possível saltar registros;
- Normalmente usado em fitas magnéticas;
- Primeiros S.O.

- **Acesso Aleatório: (sistemas atuais - Discos)**

- Bytes/Registros lidos em qualquer ordem;
- Leitura pode ser:
 - **move o marcador** do arquivo (Seek), e então **read...**
 - Lê e move o marcador de arquivo;
- Essencial para Sistemas de Base de Dados;

Gerenciamento de Arquivos

Arquivos

- **Atributos dos Arquivos:**

Attribute	Meaning
Protection	Who can access the file and in what way
Password	Password needed to access the file
Creator	ID of the person who created the file
Owner	Current owner
Read-only flag	0 for read/write; 1 for read only
Hidden flag	0 for normal; 1 for do not display in listings
System flag	0 for normal files; 1 for system file
Archive flag	0 for has been backed up; 1 for needs to be backed up
ASCII/binary flag	0 for ASCII file; 1 for binary file
Random access flag	0 for sequential access only; 1 for random access
Temporary flag	0 for normal; 1 for delete file on process exit
Lock flags	0 for unlocked; nonzero for locked
Record length	Number of bytes in a record
Key position	Offset of the key within each record
Key length	Number of bytes in the key field
Creation time	Date and time the file was created
Time of last access	Date and time the file was last accessed
Time of last change	Date and time the file has last changed
Current size	Number of bytes in the file
Maximum size	Number of bytes the file may grow to

Gerenciamento de Arquivos

Arquivos

- **Operações sobre arquivos:**

1. **Create:**

- Arquivo criado sem dados;
- Anunciar que o arquivo existe e definir atributos;

2. **Delete:**

- Libera espaço em disco;
- Chamada de sistema para esse fim.

3. **Open:**

- Permitir que o S.O busque e coloque na memória principal os atributos e lista de endereços do disco, para tornar mais rápido o acesso das chamadas posteriores;

Gerenciamento de Arquivos

Arquivos

- **Operações sobre arquivos:**

- 4. Close**

- Libera espaço na tabela interna;
 - Atributos e endereços do disco;
 - Se um disco é escrito em blocos, o fechamento força a escrita do ultimo bloco do arquivo, mesmo se não estiver completo;

- 5. Read**

- Ler dados do arquivo
 - Quem faz a leitura deve especificar a posição de leitura, quantidade de dados a ser lido e o buffer para colocar os dados;

Gerenciamento de Arquivos

Arquivos

- **Operações sobre arquivos:**

- 6. Write**

- Escrita em uma posição determinada;
 - Se posição for final de arquivo, o arquivo sofrerá um aumento de tamanho;
 - Posição no meio: dados subscritos;

- 7. Append:**

- Adiciona dados no final do arquivo;

- 8. Seek**

- Reposiciona o ponteiro do arquivo para um local específico do arquivo;
 - Os dados são lidos/escritos da posição determinada

Gerenciamento de Arquivos

Arquivos

- **Operações sobre arquivos:**

- 9. Get Attributes:**

- Retorna os atributos do arquivo;
 - Utilizados por muitos programas;

- 10. Set Attributes:**

- Permite a alteração, pelo usuário, dos atributos de arquivo já criado;

- 11. Rename:**

- Renomear o nome do arquivo existente;

Gerenciamento de Arquivos

Arquivos

- Exemplo de um programa com chamada para arquivos

```
/* File copy program. Error checking and reporting is minimal. */

#include <sys/types.h> /* include necessary header files */
#include <fcntl.h>
#include <stdlib.h>
#include <unistd.h>

int main(int argc, char *argv[]); /* ANSI prototype */

#define BUF_SIZE 4096 /* use a buffer size of 4096 bytes */
#define OUTPUT_MODE 0700 /* protection bits for output file */

int main(int argc, char *argv[])
{
    int in_fd, out_fd, rd_count, wt_count;
    char buffer[BUF_SIZE];

    if (argc != 3) exit(1); /* syntax error if argc is not 3 */

    /* Open the input file and create the output file */
    in_fd = open(argv[1], O_RDONLY); /* open the source file */
    if (in_fd < 0) exit(2); /* if it cannot be opened, exit */
    out_fd = creat(argv[2], OUTPUT_MODE); /* create the destination file */
    if (out_fd < 0) exit(3); /* if it cannot be created, exit */
```

Descritor de arquivos: valor inteiro criado pelo SO no momento da criação ou abertura de um arquivo. Todas as operações sobre o arquivo são realizadas por meios deste numero.

Gerenciamento de Arquivos

Arquivos

- Exemplo de um programa com chamada para arquivos

```
/* Copy loop */
while (TRUE) {
    rd_count = read(in_fd, buffer, BUF_SIZE); /* read a block of data */
    if (rd_count <= 0) break;                  /* if end of file or error, exit loop */
    wt_count = write(out_fd, buffer, rd_count); /* write data */
    if (wt_count <= 0) exit(4);                /* wt_count <= 0 is an error */
}
```

```
/* Close the files */
close(in_fd);
close(out_fd);
if (rd_count == 0) /* no error on last read */
    exit(0);
else
    exit(5);       /* error on last read */
```

Gerenciamento de Arquivos

Arquivos - Arquivos Mapeados em Memória

- Mapear arquivos no espaço de endereçamento do processo em execução:
 - novas chamadas de sistema:
 - **map**: fornece um nome de arquivo e um endereço virtual – S.O mapea em um espaço de endereçamento virtual;
 - **unmap**: desmapea o arquivo virtual;
 - Quando o processo termina, o arquivo é deixado no disco, como se tivesse sido alterado por uma combinação de chamadas: seek e write;

Gerenciamento de Arquivos

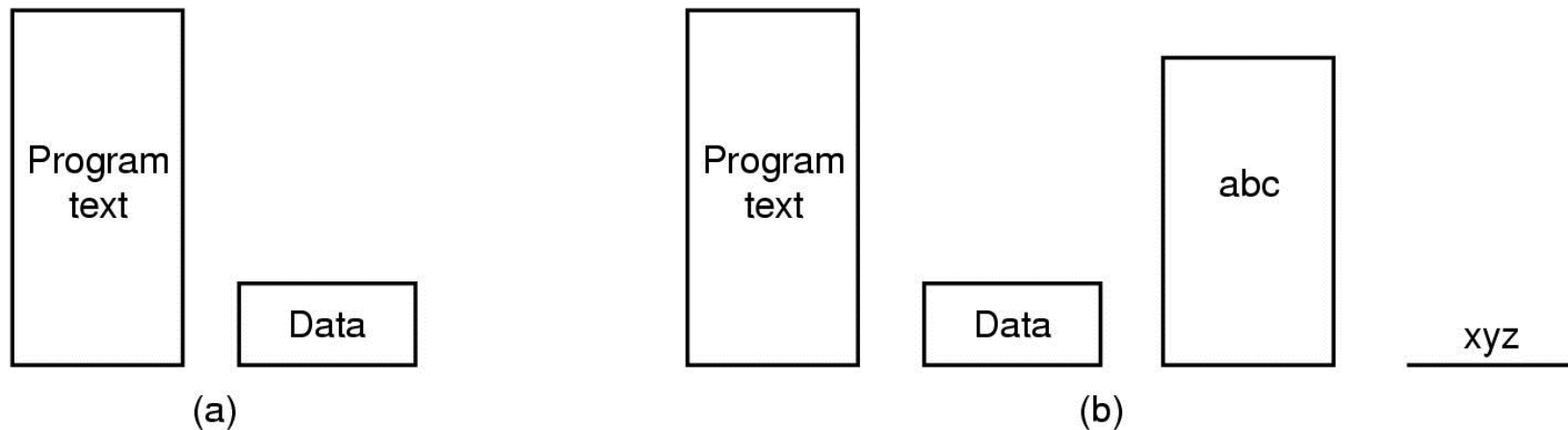
Arquivos - Arquivos Mapeados em Memória

- Mapear arquivos no espaço de endereçamento do processo em execução:
 - Funcionamento:
 - **Exemplo:** mapear um arquivo de 64 K a partir do endereço 512 K
 - As **tabelas internas do sistema** são alteradas para que o arquivo se torne um armazenamento sobressalente da região 512 K a 576 K
 - Uma leitura no endereço 512 K causa uma **falta de página**, a qual é carregada para aquela posição;
 - Uma vez que o processo termine, todas as paginas mapeadas e modificadas são **escritas de volta** em seu arquivos;

Gerenciamento de Arquivos

Arquivos - Arquivos Mapeados em Memória

- Mapear arquivos no espaço de endereçamento do processo em execução:
 - (a) processo segmentado antes de mapear arquivos em se espaço de endereçamento;



- (b) processo depois de mapear um arquivo *abc* existente em um segmento e criando um novo para *xyz*²⁶

Gerenciamento de Arquivos

Arquivos - Arquivos Mapeados em Memória

- Mapear arquivos no espaço de endereçamento do processo em execução:
 - **Vantagem:**
 - Elimina a necessidade de operações E/S contínua a dispositivos externos de armazenamento: disco;
 - **Desvantagens/Problemas:**
 - Sistema não sabe o tamanho exato do arquivo de saída ou quantos bytes foram escritos naquela página;
 - o S.O consegue apenas criar um arquivo do tamanho da página;

Gerenciamento de Arquivos

Arquivos - Arquivos Mapeados em Memória

- Mapear arquivos no espaço de endereçamento do processo em execução:

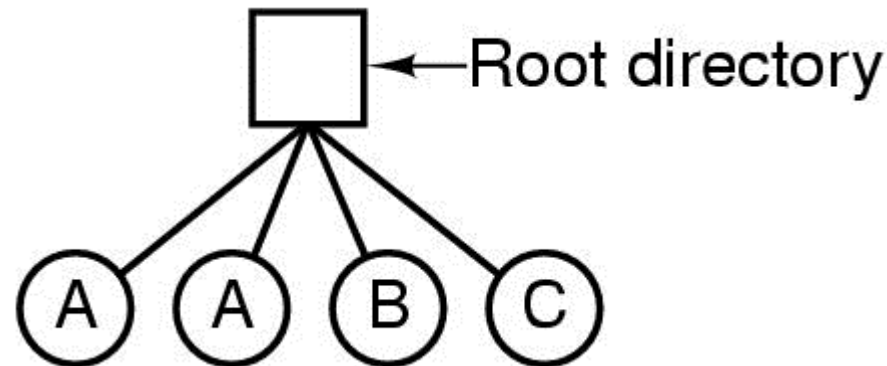
- **Desvantagens/Problemas:**

- Arquivo mapeado por um processo e aberto para leitura por outro processo
 - Alteração só é realizada quando a página é retirado da memória – versões inconsistentes;
- Tamanho do arquivo maior que o espaço de endereçamento;
 - Solução: providenciar para que a chamada de sistema *map* possa mapear uma parte do arquivo e não apenas todo o arquivo;

Gerenciamento de Arquivos

Diretórios

- **Sistemas de Diretórios de Nível Simples:**

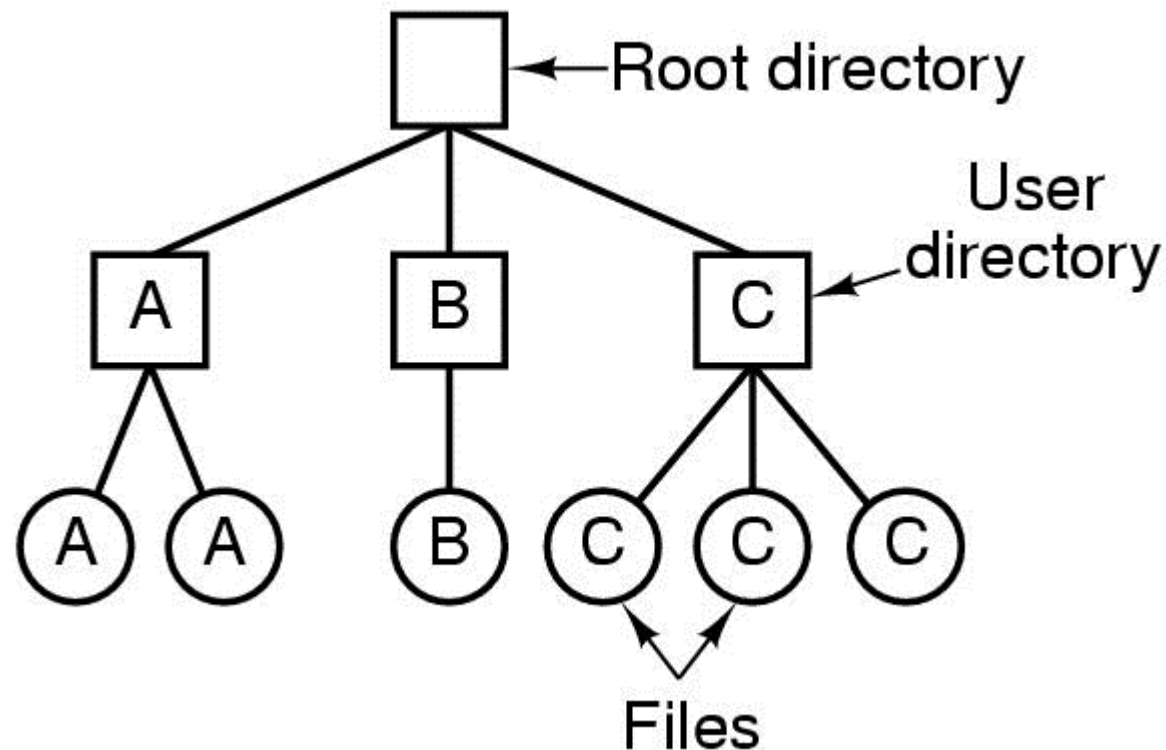


- **Um sistema de diretório de nível simples:**
 - contêm 4 arquivos;
 - 3 donos diferentes;
- **Supercomputador CDC 6600 utiliza esse tipo de diretórios;**
- **Vantagem:** simplicidade e a capacidade de localizar os arquivos rapidamente;
- **desvantagens:** nomes com arquivos iguais;

Gerenciamento de Arquivos

Diretórios

- **Sistemas de Diretórios de Dois Níveis:**

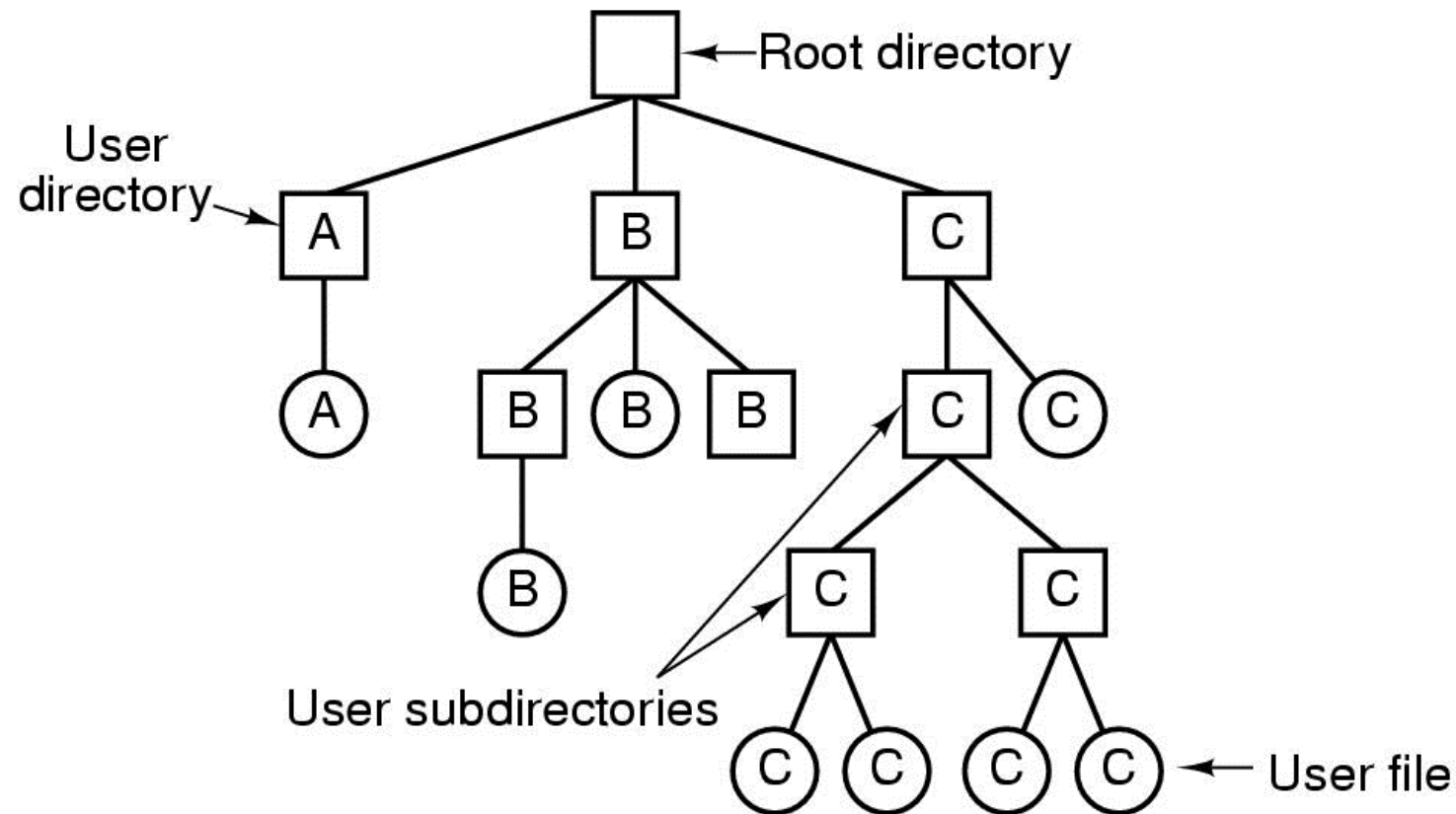


- **Letras indicam o proprietário do Diretório e do Arquivo**

Gerenciamento de Arquivos

Diretórios

- **Sistemas de Diretórios Hierárquico:**



Gerenciamento de Arquivos

Diretórios

- **Nomes de Caminhos:**
 - Dois métodos:
 - Nome do Caminho Absoluto
 - Sempre iniciam no diretório raiz e são únicos;
 - No Unix, separados por /
 - No Windows, separados por \
 - No MULTICS, separados por >
 - Se começar com separador é absoluto
 - /usr/ast/caixapostal

Gerenciamento de Arquivos

Diretórios

- **Nomes de Caminhos:**

- Dois métodos:

- Nome de Caminho Relativo:

- Usado juntamente com o conceito de **Diretório de Trabalho ou Atual;**

- Todos os nomes de caminho que não comecem no diretório raiz são assumidos como relativos ao diretório de trabalho;

- Ex: Fazem a mesma coisa

- `cp /usr/ast/caixapostal /usr/ast/caixapostal.bak`

- `cp caixapostal caixapostal.bak`

Gerenciamento de Arquivos

Diretórios

- **Nomes de Caminhos:**

- A maioria dos S.Os suportam um sistema de diretório hierarquico com duas entradas especiais em cada diretório:

- `'.'` - diretório atual

- `'..'` - diretório pai

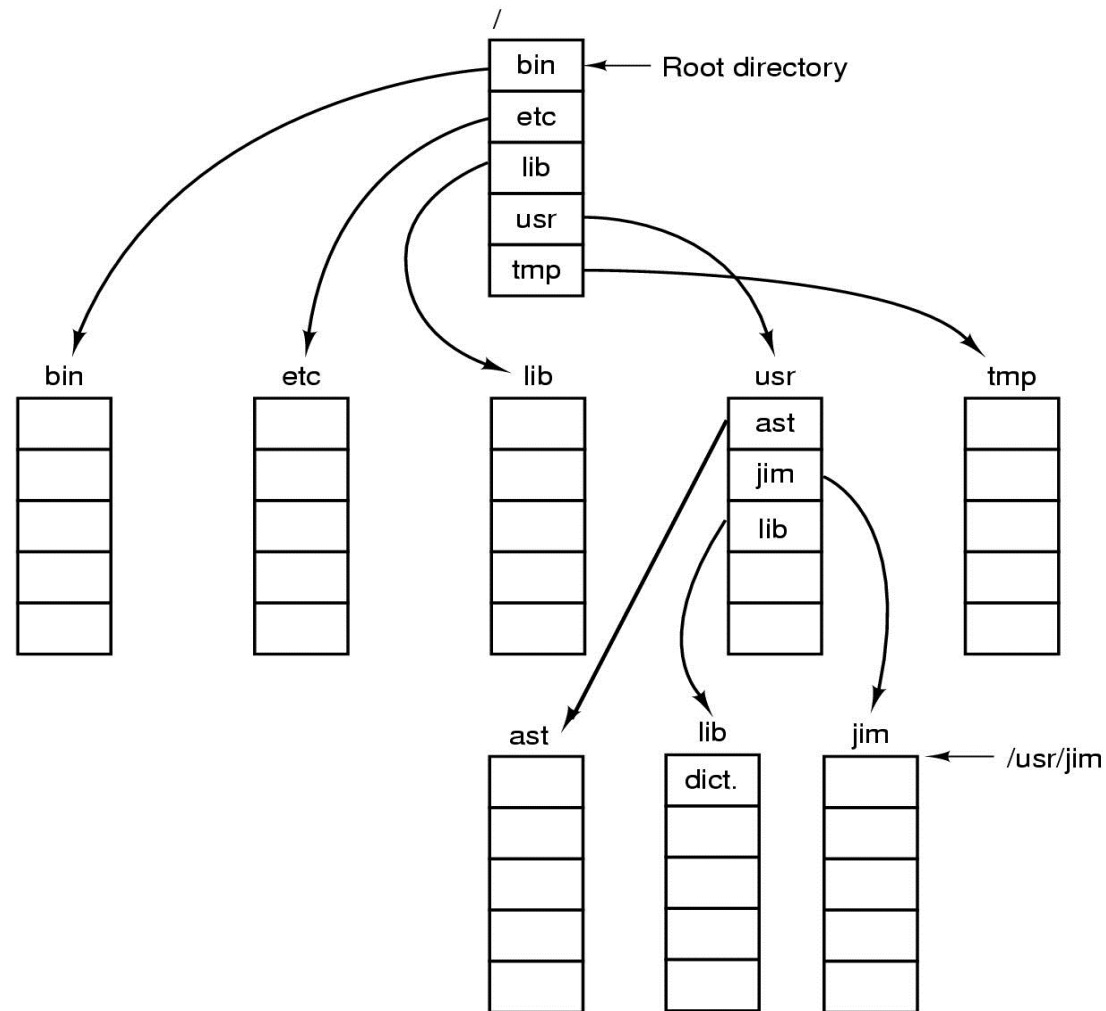
- **Ex:**

- `cp ../bib/dicionario .` (copia o arquivo para o seu diretório)

Gerenciamento de Arquivos

Diretórios

- **Sistemas de Diretório em Árvore do UNIX**



Gerenciamento de Arquivos

Diretórios

- **Operações sobre Diretórios**

- 1. Create**

- Cria diretório vazio, apenas com '.' e '..'

- 2. Delete**

- remove diretório, se estiver vazio.

- 3. Opendir**

- Lê diretórios. Antes de ler deve ser aberto, como em arquivos;

- 4. Closedir**

- quando acaba de ser lido, ele é fechado para liberar espaço na tabela interna;

Gerenciamento de Arquivos

Diretórios

- Operações sobre Diretórios

5. Readdir

- Retorna a próxima entrada em um diretório aberto
- Criado para padronizar, pois antes se usava **read** de arquivo, forçando o usuário lidar e conhecer as estruturas internas dos diretórios;

6. Rename

- Renomear o nome de um diretório;

Gerenciamento de Arquivos

Diretórios

- **Operações sobre Diretórios**

7. Link

- Técnica que permite que um arquivo aparece em mais de um diretório.
- Para cada link é incrementado um contador no i-node do arquivo - conhecida com ligação estrita – hard link

8. Unlink

- Remove a entrada de um arquivo em um diretório;
- Se o arquivo estiver em apenas 1 diretório, ele é removido;
- Se estiver em vários, apenas o nome será removido;
- Unix utiliza link e unlink.

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

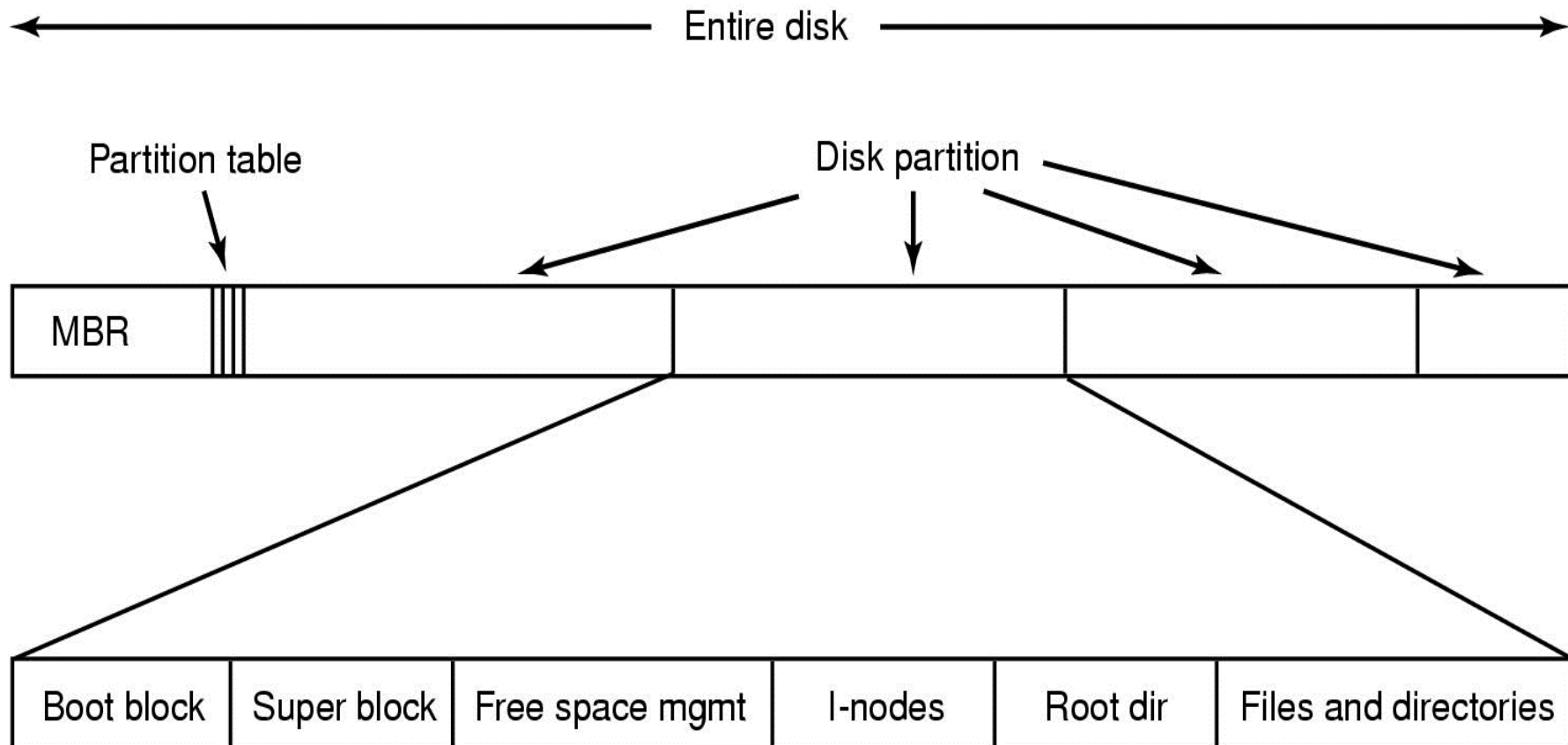
- **Esquema do Sistema de Arquivos**

- Os discos são divididos em uma ou mais partições, com sistemas de arquivos independentes para cada uma;
- Setor 0 do disco
 - Chamada de **MBR** – Master Boot Record
 - usado para iniciar o computador
 - Fim da MBR contem a tabela de partição
 - Endereços iniciais e finais de cada partição
 - Uma das partições é marcada como ativa;
 - Inicialização – BIOS lê MBR – partição ativa
 - lê o primeiro bloco – **bloco de boot**
 - **carrega o S.O. que está na partição ativa**⁴⁰

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Esquema do Sistema de Arquivos**
 - **Possível layout de um sistema de arquivo**



Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

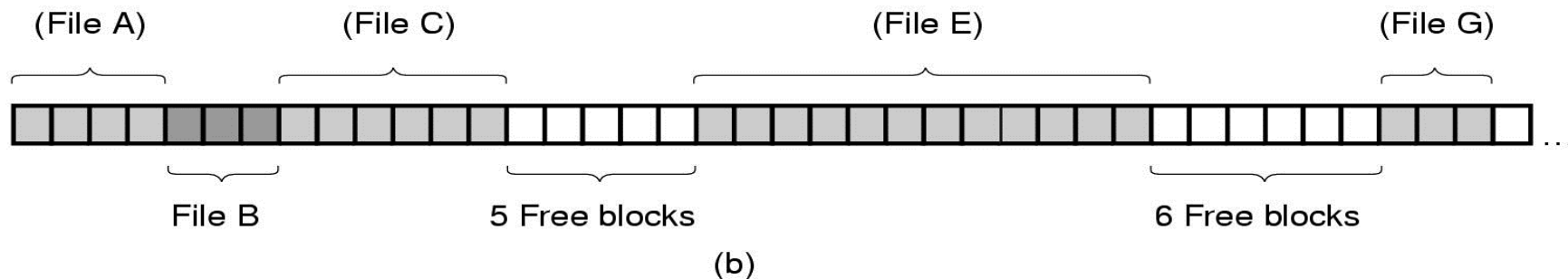
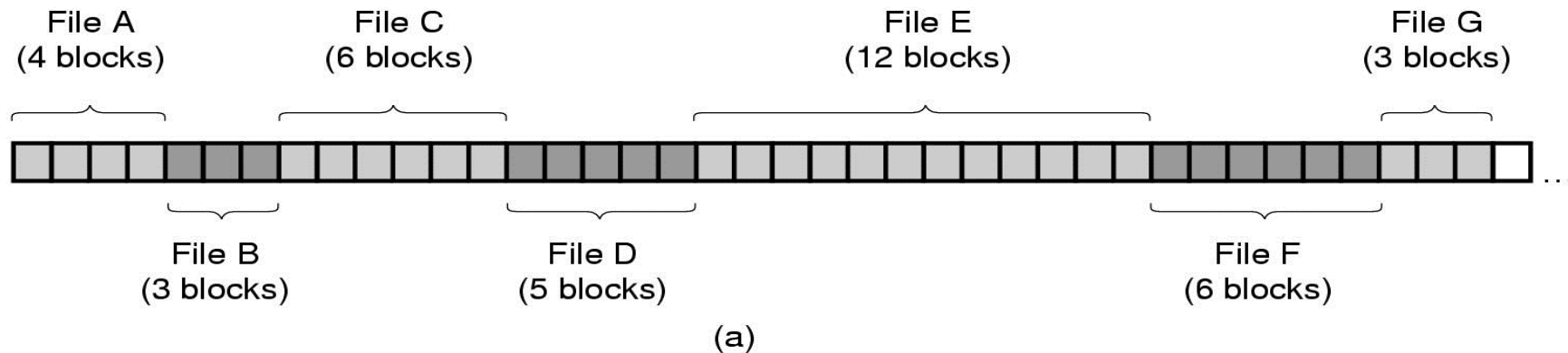
Implementação de Arquivos

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**

- (a) **Alocação Contígua** de 7 arquivos
- (b) Depois dos arquivos D e F serem removidos



Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**

- **Alocação Contígua**

- **Vantagens:**

- Simples de Implementar

- lembrar dois números: endereço em disco do primeiro bloco e numero de blocos no arquivo

- Desempenho de leitura é excelente

- todo o arquivo pode ser lido em uma única operação;
 - Basta realizar um Seek inicial e realizar a leitura;

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**

- **Alocação Contígua**

- **Desvantagens:**

- Com o tempo o disco fica fragmentado;
 - Custo para se fazer a realocação dos blocos do arquivos;

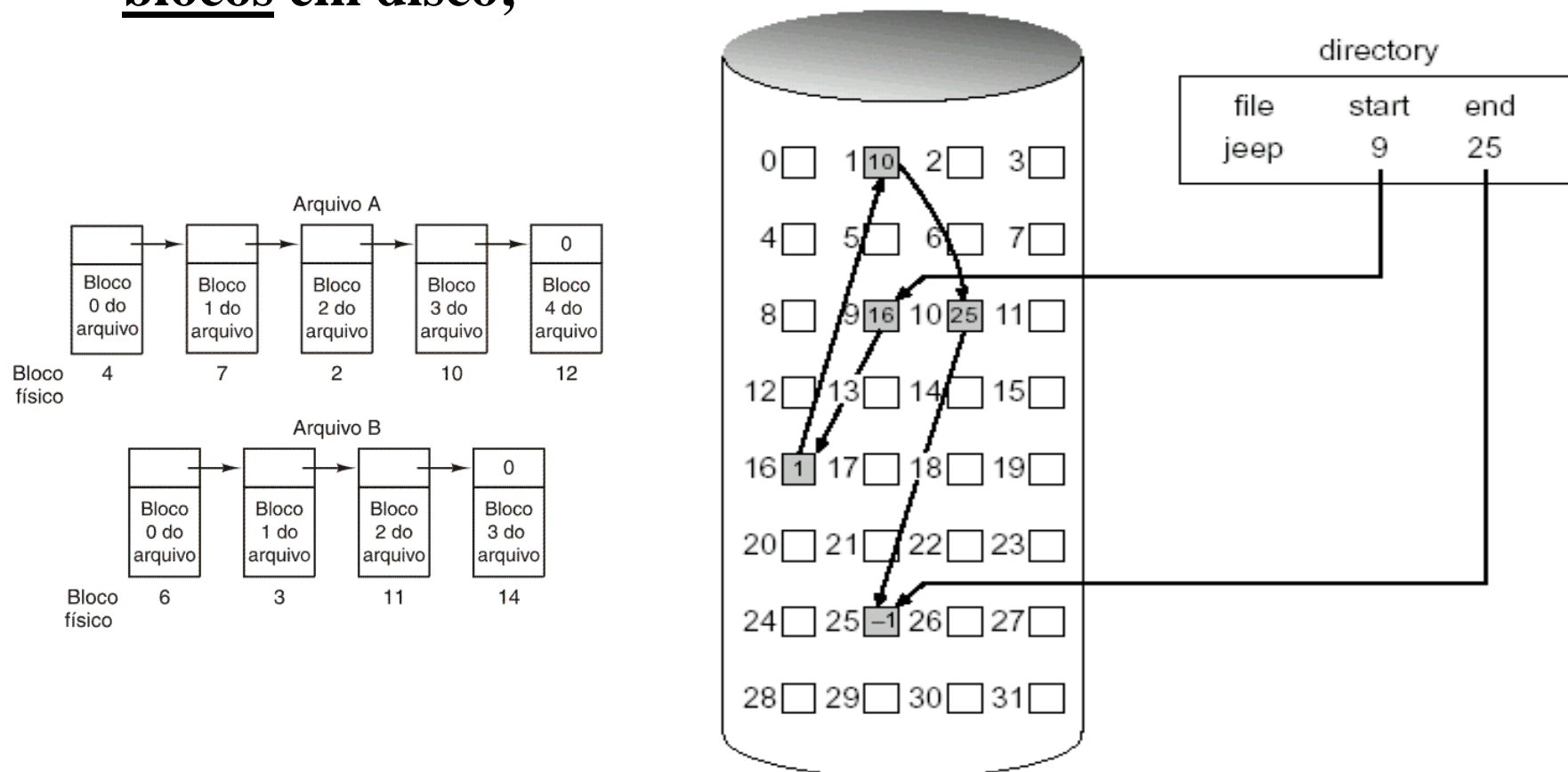
- Amplamente utilizadas em CD-ROMS e DVDs

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**

- Armazenamento de um arquivo como uma lista ligada de blocos em disco;



Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**
- **Lista ligada de blocos em disco;**
 - **Vantagens:**
 - Não há fragmentação;
 - manter apenas uma entrada de diretório para o endereço do primeiro bloco;
 - **Desvantagens:**
 - Acesso aleatório extremamente lento;
 - para chegar ao bloco n deve passar pelos $n-1$
 - Quantidade de dados não é mais potência de 2, devido ao ponteiro de próximo; (programas leem potência de 2)
 - Requer obter e concatenar dois blocos, gerando sobrecarga extra;

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**

- Alocação por lista ligada usando uma tabela alocação de arquivo em RAM

- Elimina ponteiro no bloco

Physical
block

0	
1	
2	10
3	11
4	7
5	
6	3
7	2
8	
9	
10	12
11	14
12	-1
13	
14	-1
15	

Tabela
FAT

← File A starts here

← File B starts here

← Unused block

Gerenciamento de Arquivos

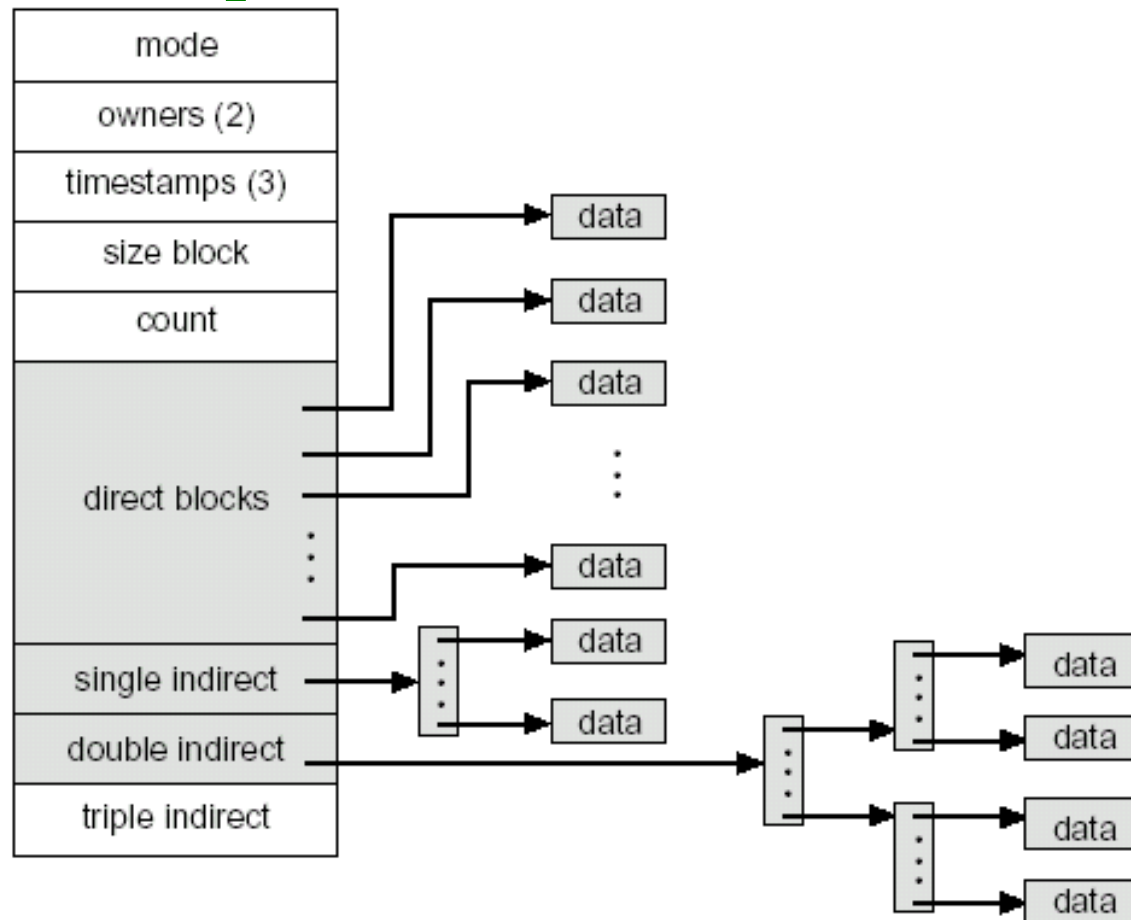
Implementação do Sistema de Arquivos

- **Implementação de Arquivos**
- Alocação por lista ligada usando uma tabela alocação de arquivo em RAM
 - **Vantagens**
 - Elimina ponteiro no bloco
 - Acesso mais rápido
 - **Desvantagens**
 - Toda a tabela deve estar na memória;
 - tabela possui entradas para todos os blocos de dados do disco

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**



- **i-Node (Unix)**
- **Tabela de Índices para cada arquivo**

Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**
- **i-Node (Unix)**
 - **Vantagens:**
 - O i-node não precisa ficar na memória inteiramente como no método anterior, fica apenas quando arquivo estiver aberto;
 - inode ocupar n bytes e no Maximo k arquivos = kn bytes
 - Leva em consideração o numero máximo de arquivo abertos em um determinado tempo

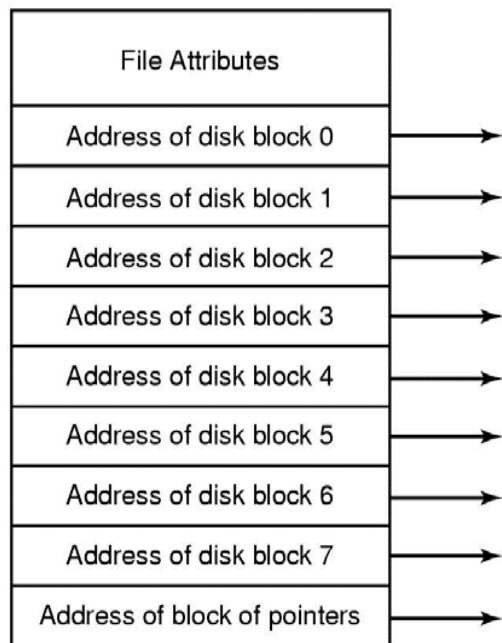
Gerenciamento de Arquivos

Implementação do Sistema de Arquivos

- **Implementação de Arquivos**

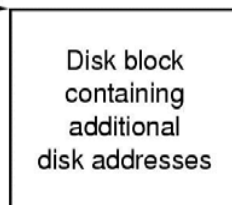
- **i-Node (Unix)**

- **Desvantagens:**



- Se a estrutura do i-node tiver espaço para um numero fixo de endereços de disco, o que acontece se o arquivo crescer?

- **Solução:** reservar o ultimo endereço para o endereço de um bloco contendo mais endereços de blocos de disco;



Gerenciamento de Arquivos

- **Referências Utilizadas:**

- Livro do Tanenbaum

- Sistemas Operacionais Modernos

- www.cs.vu.nl/~ast

- Livro do Silberschatz

- Operating System Concepts

- www.bell-labs.com/topic/books/aos-book/

- Livro do Machado e Maia

- Arquitetura de Sistemas Operacionais.