

# Aula 7

# Sistema de Arquivos

# Introdução

- Deve ser possível armazenar grande quantidade de informação
- Informação deve sobreviver ao término do processo
- Múltiplos processos podem acessar a informação concorrentemente

# Introdução

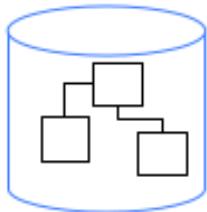
- Sistemas de Arquivos
  - Determina como os dados são armazenados
  - Gerencia o acesso ao conteúdo e aos meta-dados do arquivo
  - Torna transparente o meio físico de armazenamento
  - Garante confiabilidade e eficiência
  - Otimização e gerenciamento de espaço livre

# Introdução

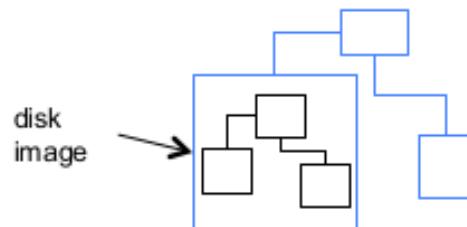
- Sistemas de Arquivos
  - Podem estar contidos em:
    - Uma partição de um disco local
    - Em uma disk image (sistema de arquivos hospedeiro)
    - Em uma partição no disco de outro computador acessível pela rede (NFS/Samba)

# Introdução

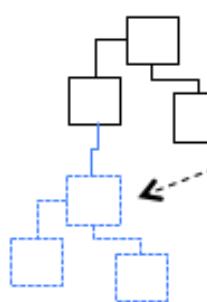
- Sistemas de Arquivos



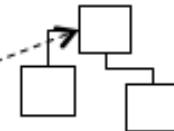
SA em uma partição



Computador

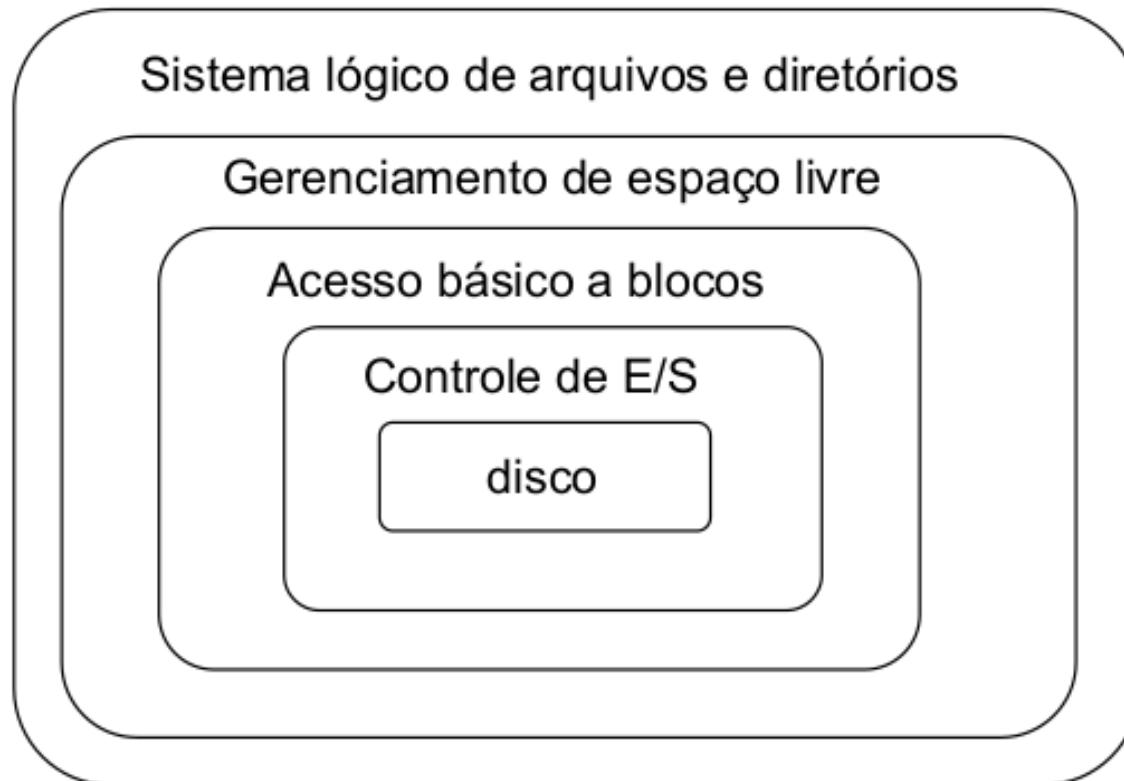


Computador A



Computador B

# Sistema de Arquivos

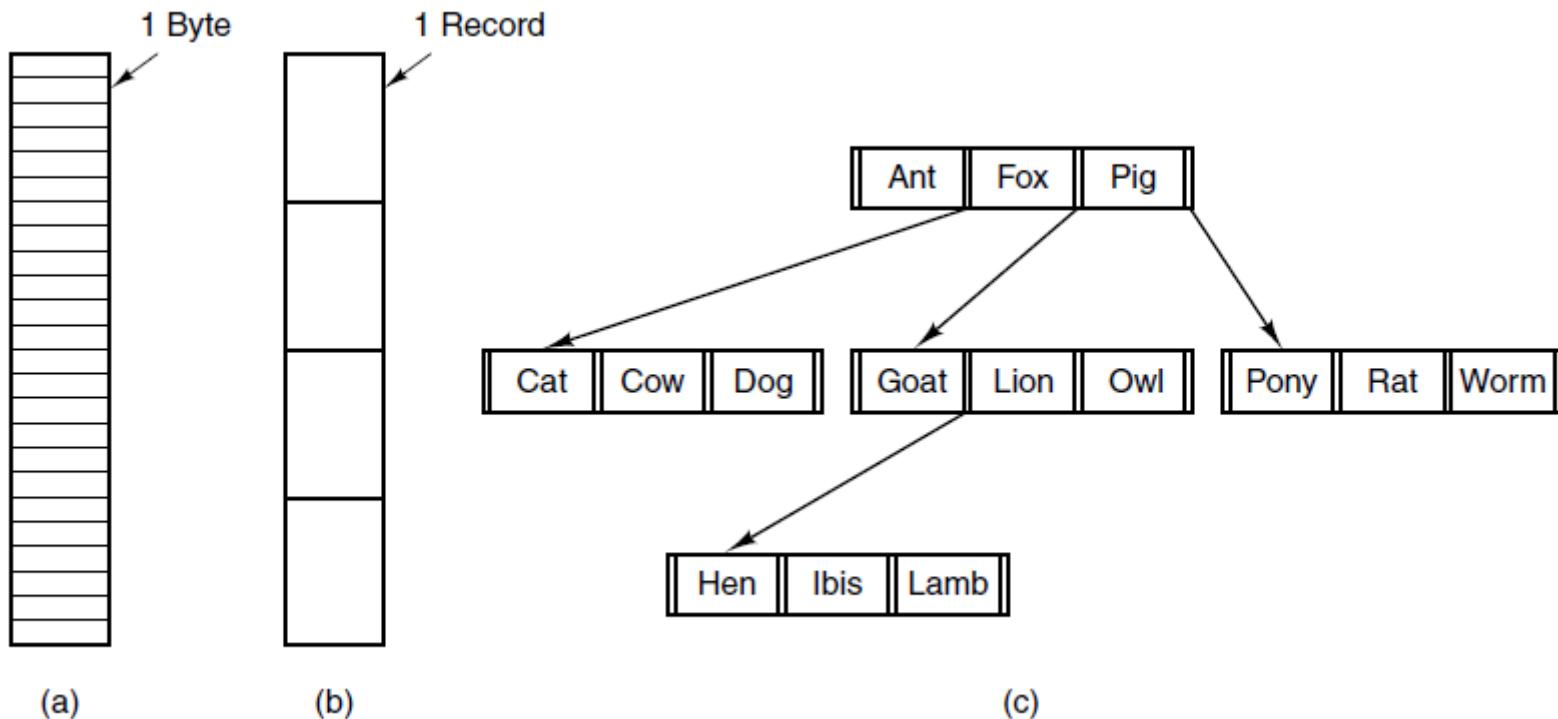


# Arquivos

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

# Arquivos

- Estruturas



# Arquivos

- Sequência de bytes:
  - Na sequencia desestruturada de bytes o SO não sabe o que o arquivo contém.
  - Qualquer significado deve ser imposto pelos programas em nível de usuário.
    - UNIX, MS-DOS e Windows utilizam essa estratégia
    - Oferece máxima flexibilidade

# Arquivos

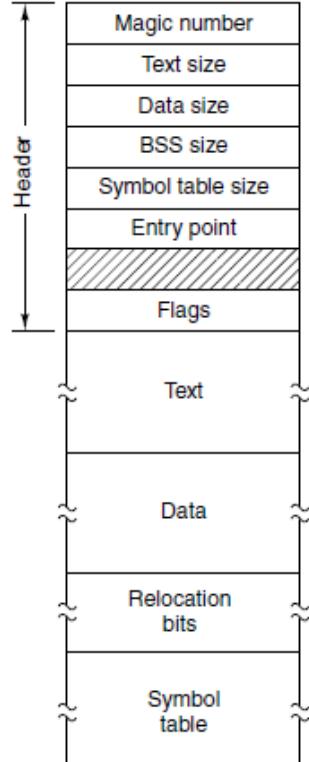
- Sequência de Registros
  - registros de tamanho fixo, cada um com alguma estrutura interna
    - operação de leitura retorna um registro e a de escrita sobrepõe ou anexa um registro
    - era comum nos computadores de grande porte na época dos cartões perfurados

# Arquivos

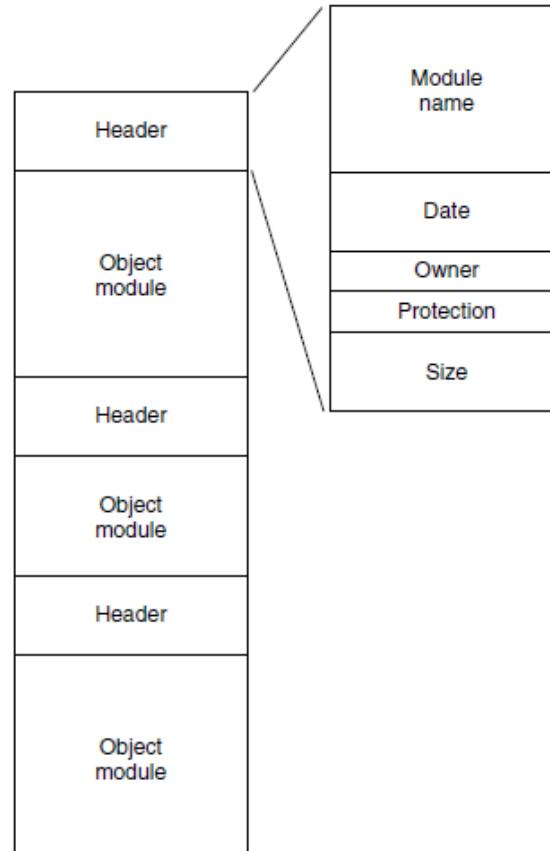
- Árvore de registros
  - campo-chave em uma posição fixa
  - a árvore é ordenada pelo campo chave pra que se busque mais rapidamente
    - aplicado em computadores de grande porte ainda usados para alguns processamentos de dados comerciais

# Arquivos

- Tipos



(a)



(b)

# Arquivos

- Atributos

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 was last changed
Current size	Number of bytes in the file
Maximum size	Number of bytes the file may grow to

# Arquivos

- Atributos

permission modes	# links	owner	group	size (bytes)	date (modified)	file name
drwxr-xr-x	2	root	root	4096	Mar 21 2002	bin
drwxr-xr-x	17	root	root	77824	Aug 11 14:40	dev
drwxr-xr-x	69	root	root	8192	Sep 25 18:15	etc
drwxr-xr-x	66	root	root	4096	Sep 25 18:15	home
dr-xr-xr-x	46	root	root	0	Aug 11 10:39	proc
drwxr-x---	12	root	root	4096	Aug 7 2002	root
drwxr-xr-x	2	root	root	8192	Mar 21 2002	sbin
drwxrwxrwx	6	root	root	4096	Sep 29 04:02	tmp
drwxr-xr-x	16	root	root	4096	Mar 21 2002	usr
-rw-r--r--	1	root	root	802068	Sep 6 2001	vmlinuz

# Arquivos

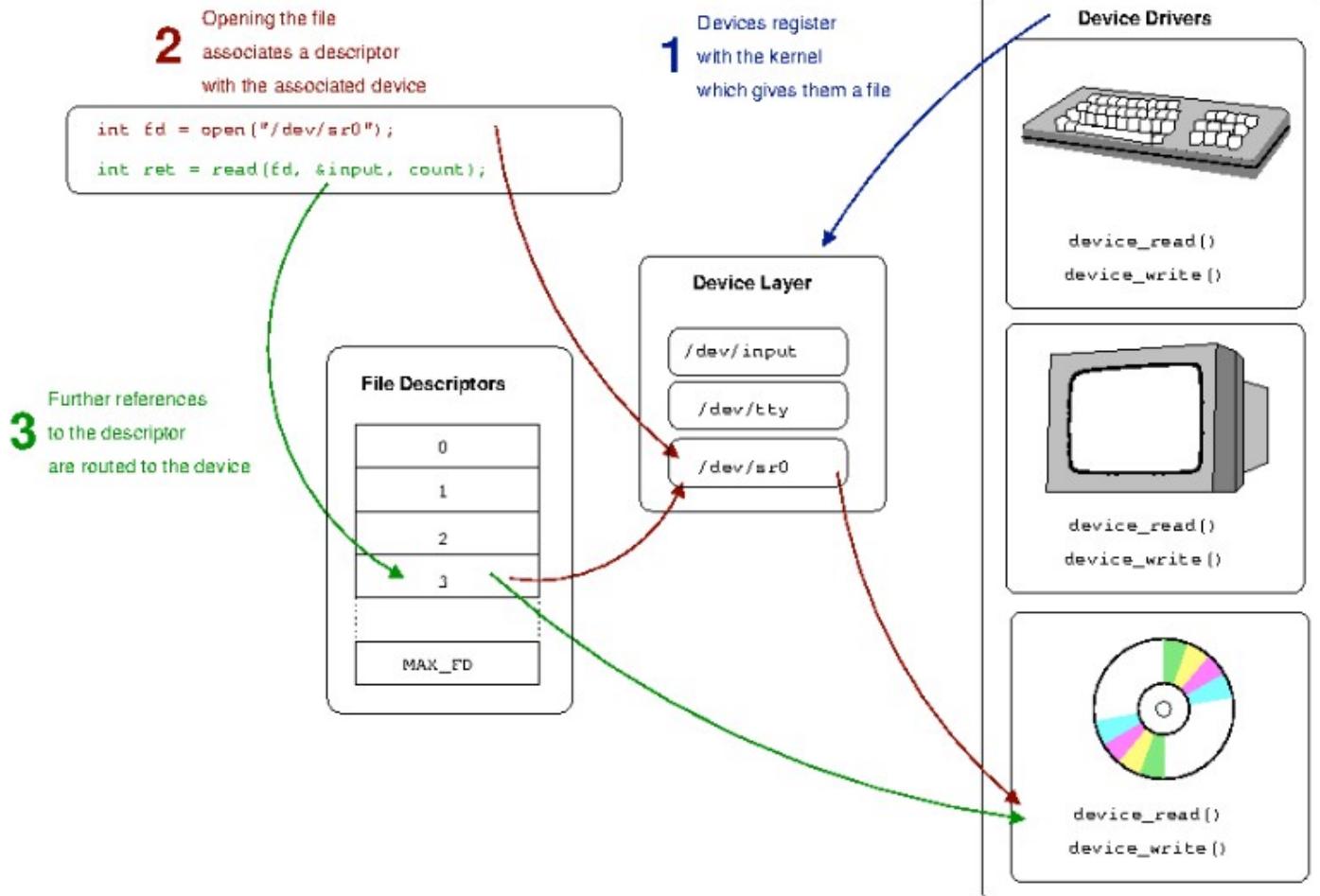
- Operações Básicas

- |           |                    |
|-----------|--------------------|
| 1. Create | 7. Append          |
| 2. Delete | 8. Seek            |
| 3. Open   | 9. Get attributes  |
| 4. Close  | 10. Set attributes |
| 5. Read   | 11. Rename         |
| 6. Write  |                    |

# Arquivos - Unix

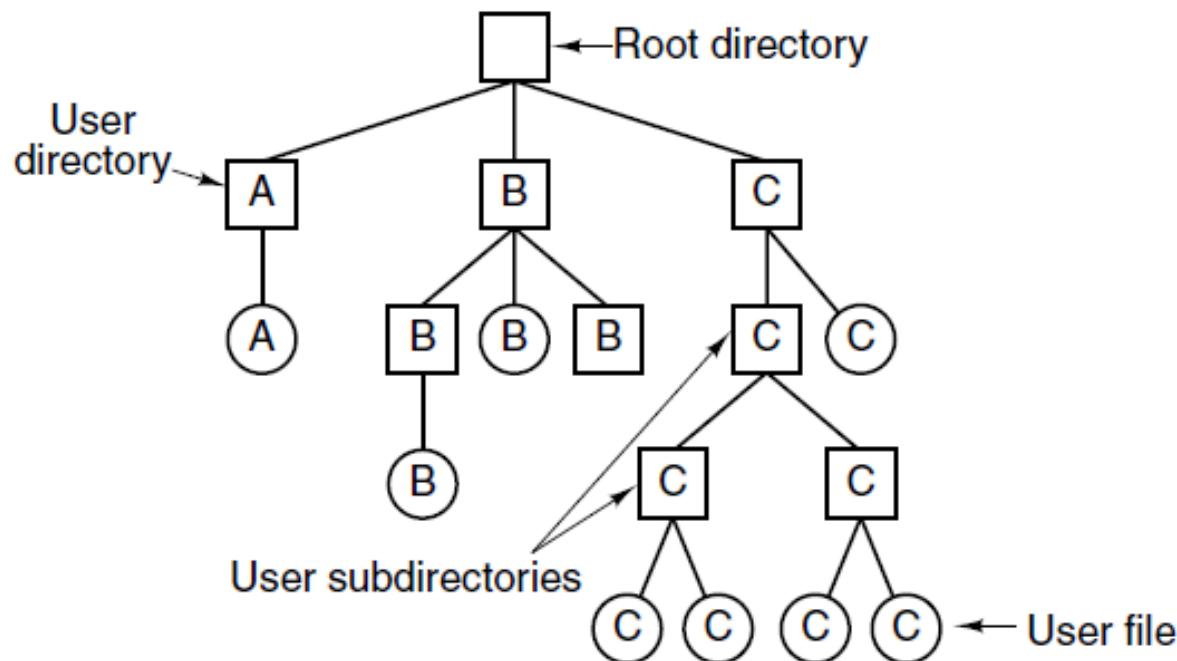
- Um sistema de arquivos UNIX é uma coleção de arquivos e diretórios com:
  - Um diretório raiz (/)
  - Cada arquivo e diretório possui identificação única:
    - seu nome e diretório em que está inserido
    - um identificador único (i-node)

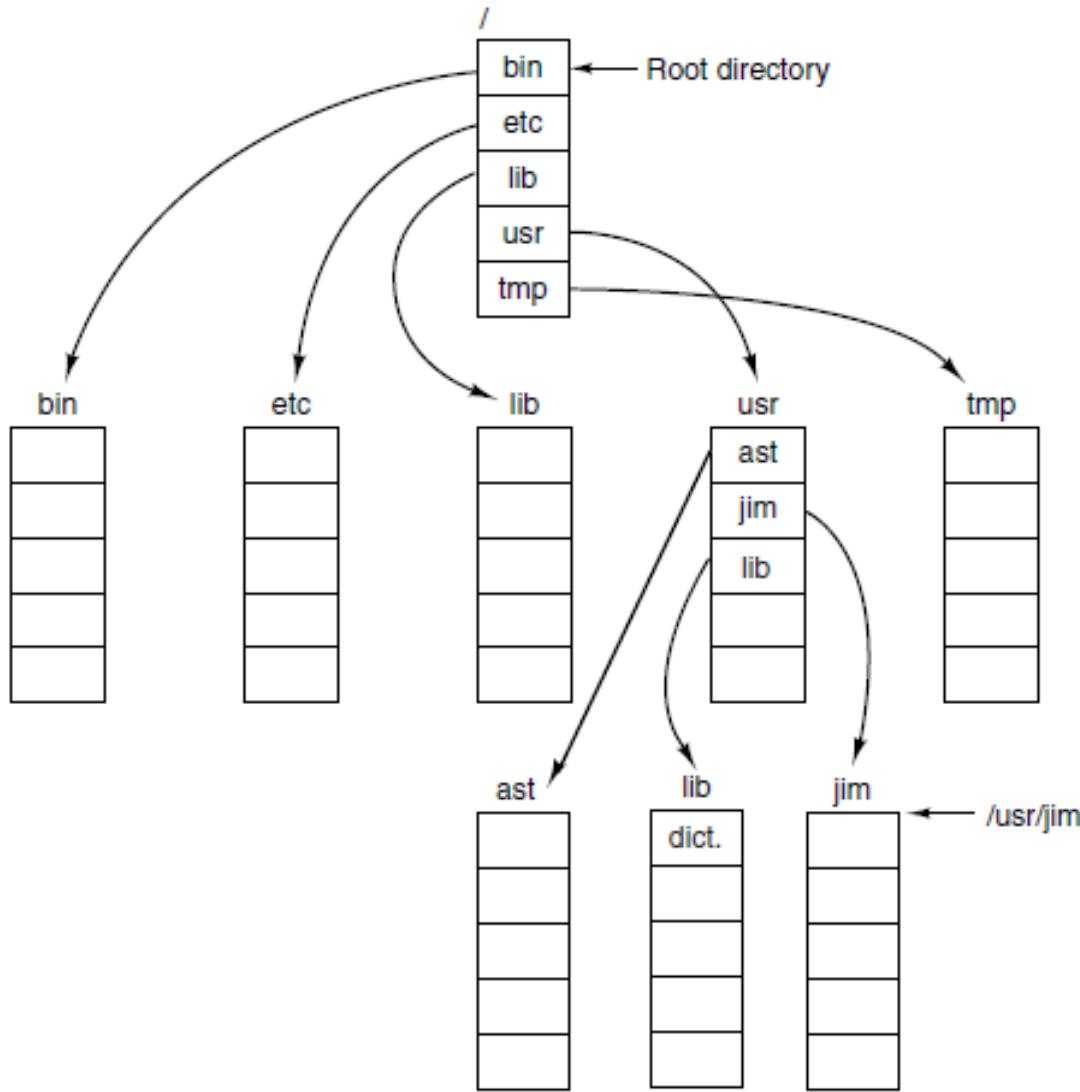
# Arquivos - Unix



# Arquivos - Unix

- Hierárquico





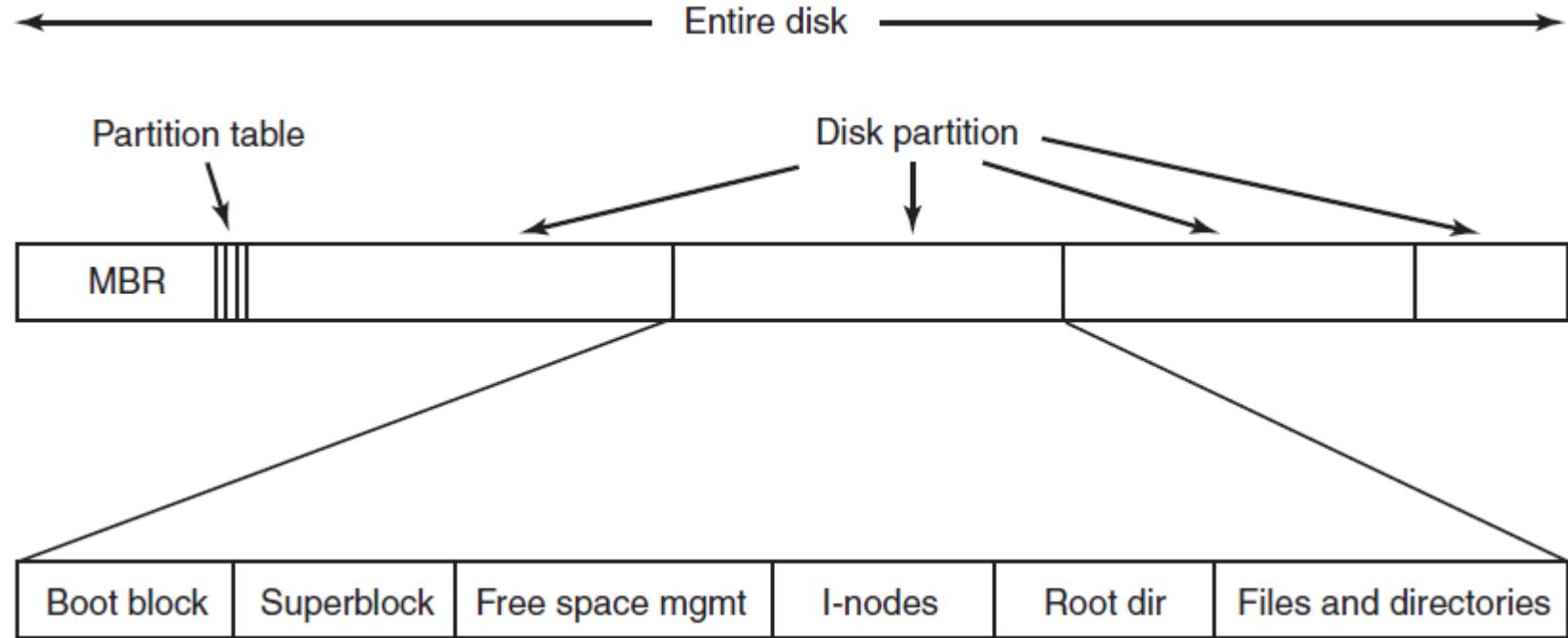


# Diretórios

- Operações básicas
  - 1. Create
  - 2. Delete
  - 3. Opendir
  - 4. Closedir
  - 5. Readdir
  - 6. Rename
  - 7. Link
  - 8. Unlink

# Layout

- Sistema de Arquivos



# Boot

- MBR → Registro Mestre de Inicialização
  - conceito de MBR foi introduzido ao público em 1983 com o PC DOS 2.0
  - contém 512 bytes de informação da estrutura organizacional do disco

# Boot

- Estrutura do MBR

Estrutura do Master Boot Record :

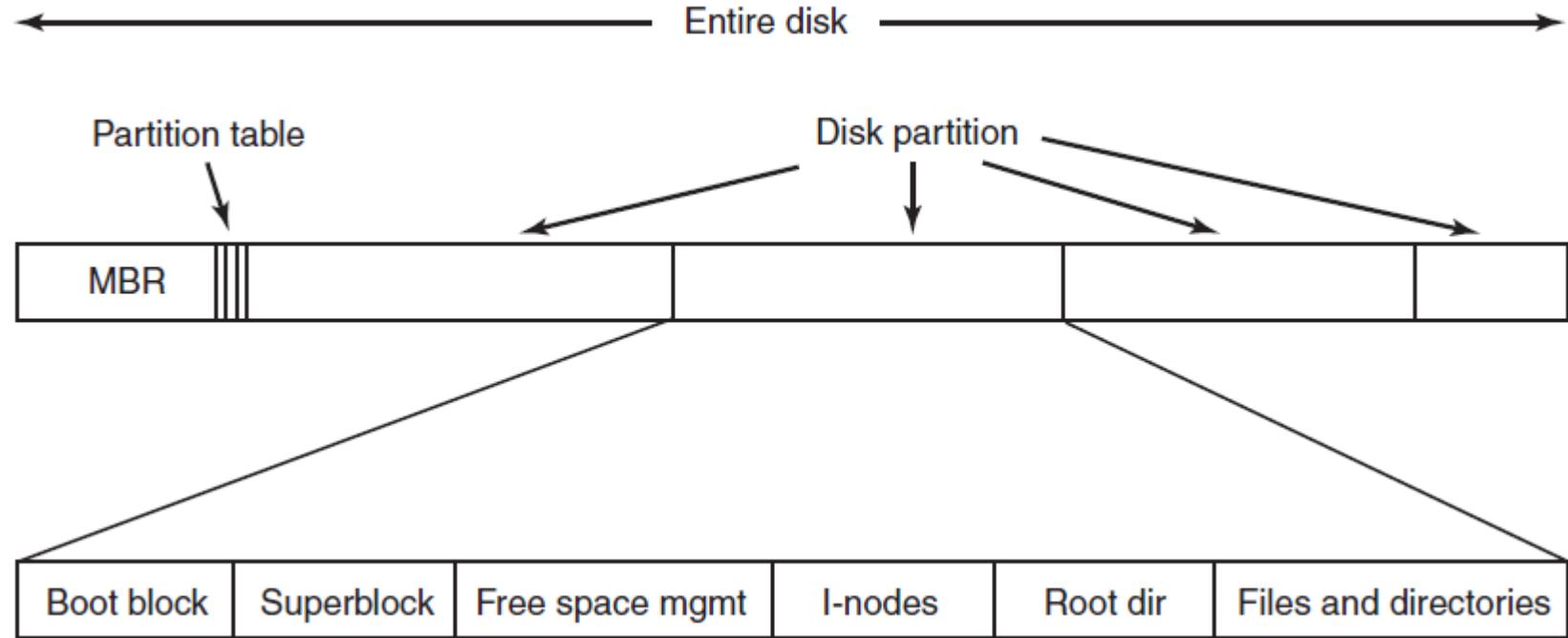
Endereço		Descrição	Tamanho em bytes
Hex	Dec		
0000	0	Código de arranque do SO	440 (max. 446)
01B8	440	Assinatura opcional	4
01BC	444	normalmente nulo ; 0x0000	2
01BE	446	<b>Tabela de partições primarias</b> (Quatro entradas de 16 bytes (IBM Partition Table scheme))	64
01FE	510	55h	2
01FF	511	AAh	
<b>Tamanho total do MBR : 440 + 4 + 2 + 64 + 2 =</b>			<b>512</b>

# Boot

- Trocar de SO
  - a MBR é sobreescrita com um novo gerenciador de boot
- Ter dois ou mais SO's instalados
  - Boot manager
    - Programa instalado na trilha MBR
      - Windows NT/2000/XP traz o NTLDR
      - Linux traz Lilo e o GRUB

# Layout

- Sistema de Arquivos

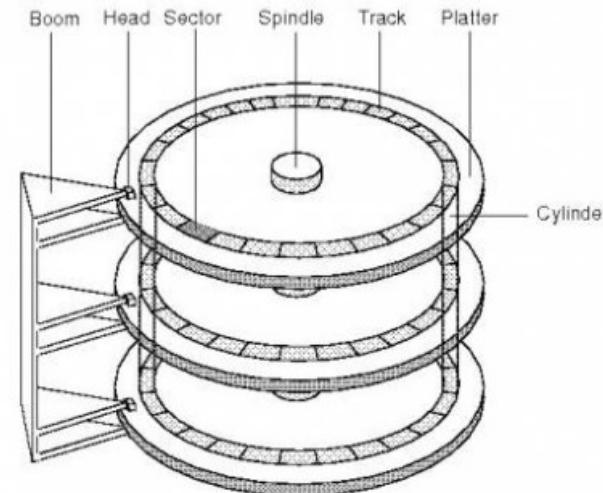
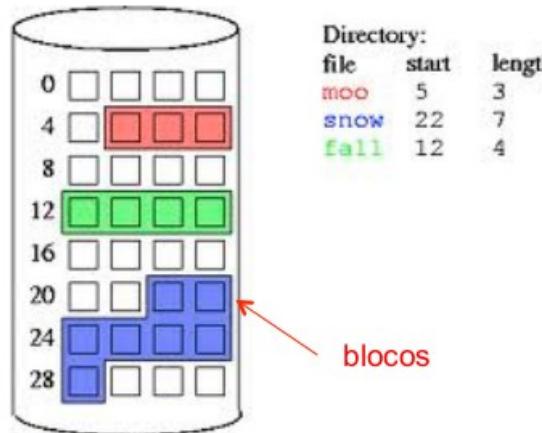


# Layout

- Boot Block
  - Informações necessárias para carregar o SO a partir desta partição
- Superblock
  - Quantidade e tamanho dos blocos
  - Contador e ponteiro de blocos livres
  - Contador e ponteiros para os FileControlBlocks (i-nodes)

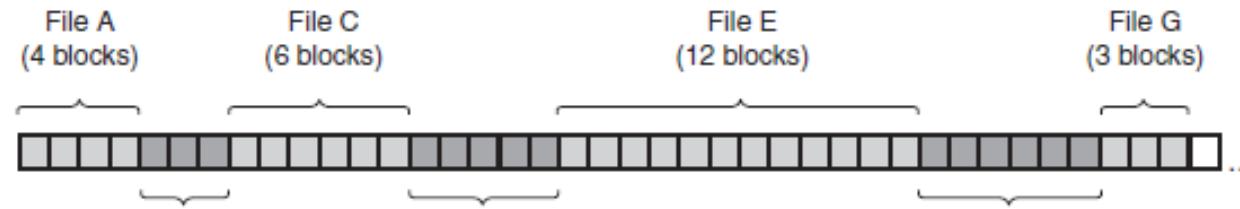
# Layout

- Arquivo é composto por uma sequência de blocos
  - Endereço lógico é traduzido para cilindro, trilha, setor

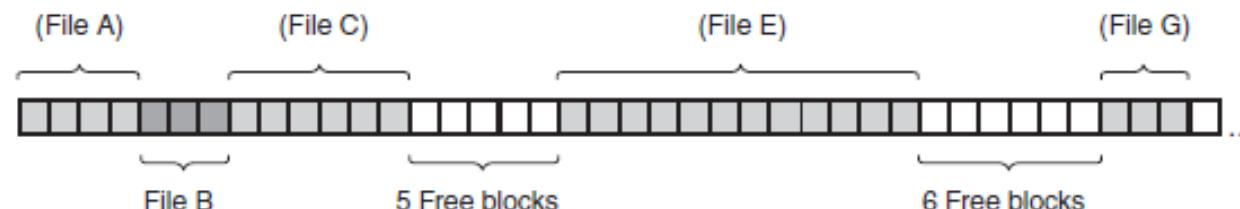


# Sistema de Arquivos

- Implementação Contínua



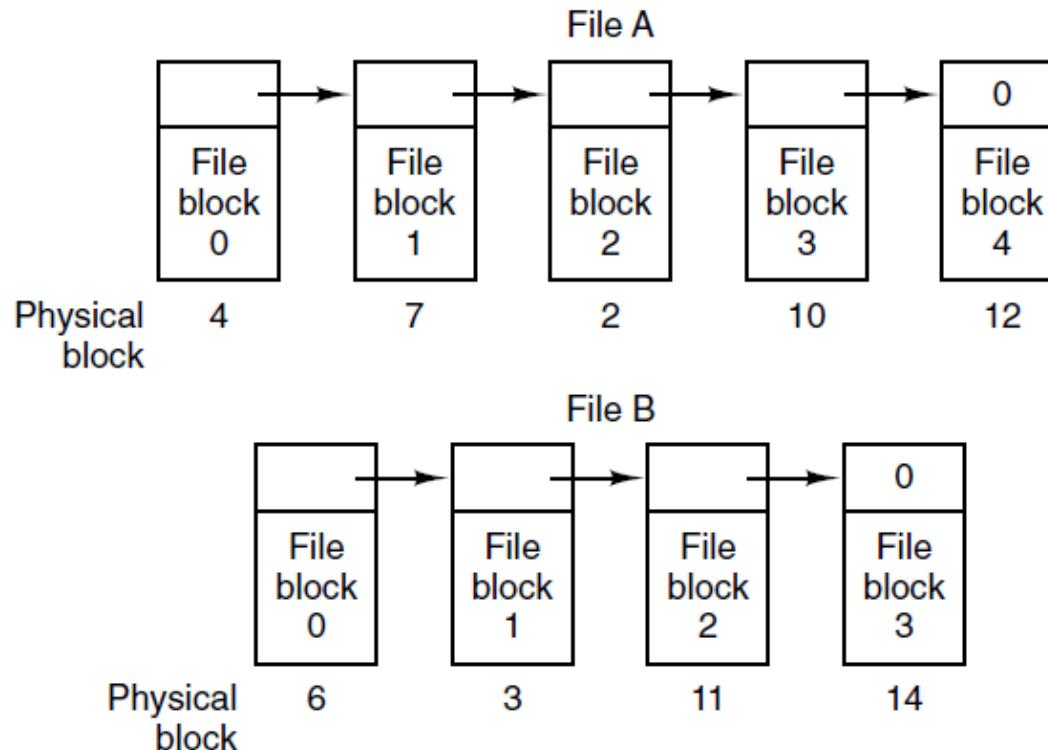
(a)



(b)

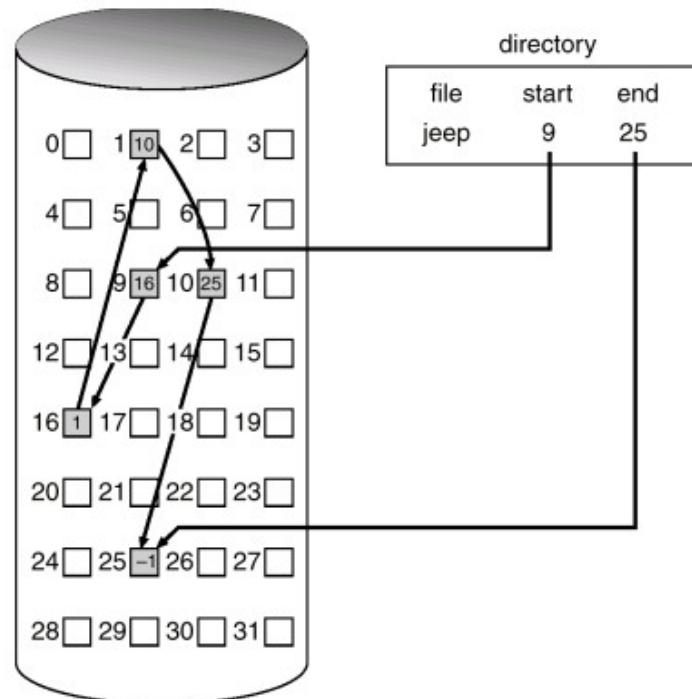
# Sistema de Arquivos

- Implementação por lista ligada



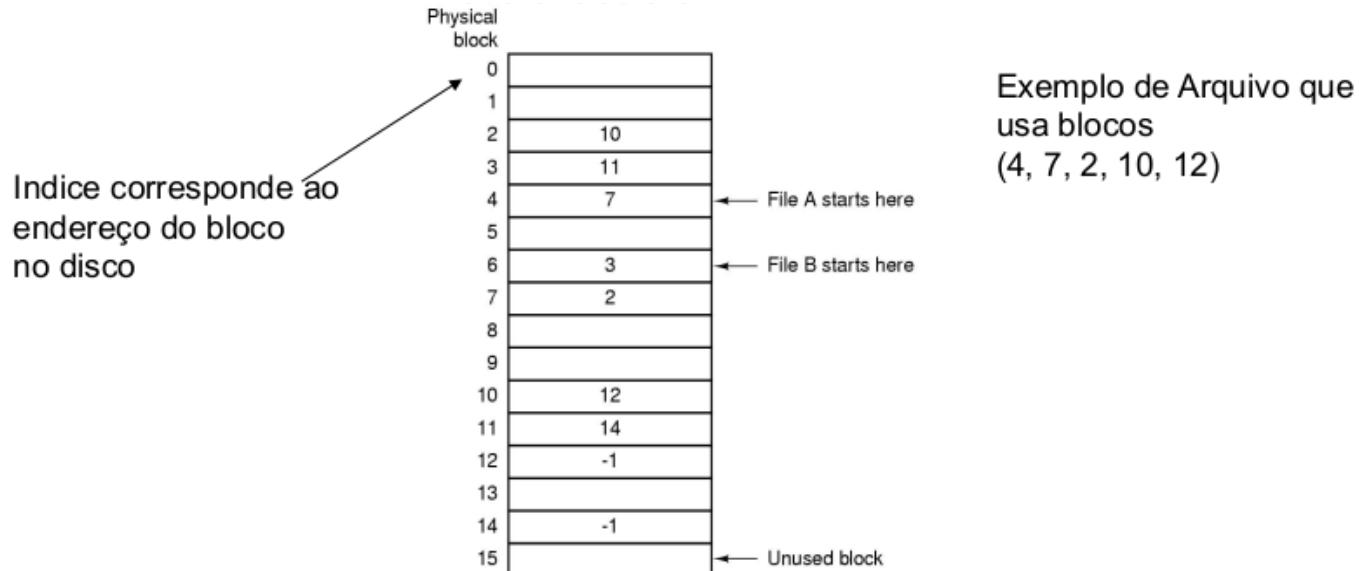
# Sistema de Arquivos

- Exemplo



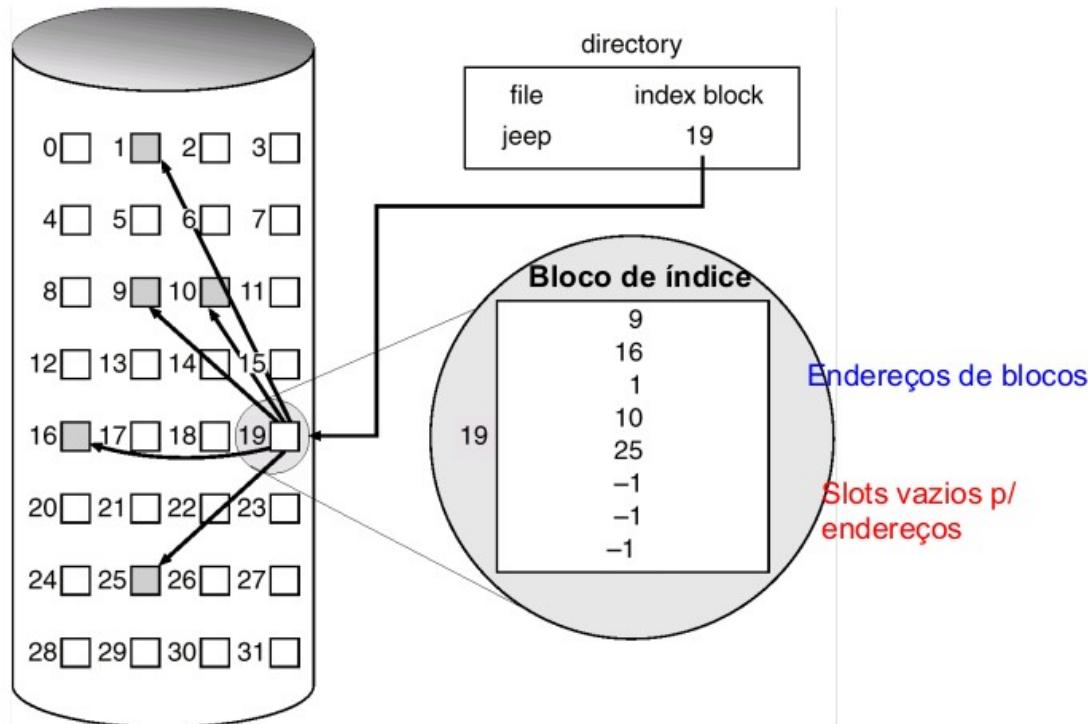
# Sistemas de Arquivos: FAT

- Lista encadeada
  - Usa tabela de alocação em memória RAM



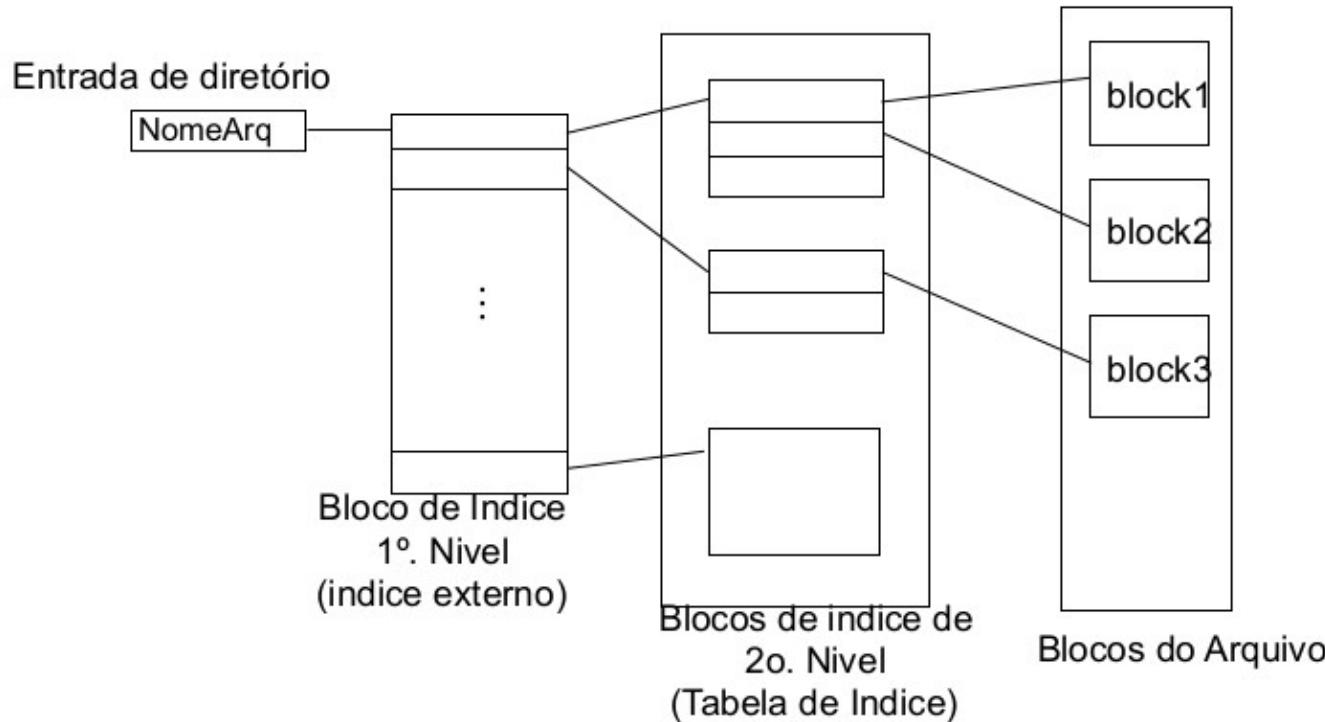
# Sistema de Arquivos: Indexado

- Bloco de index



# Sistemas de Arquivos

- Indexação multi-nível

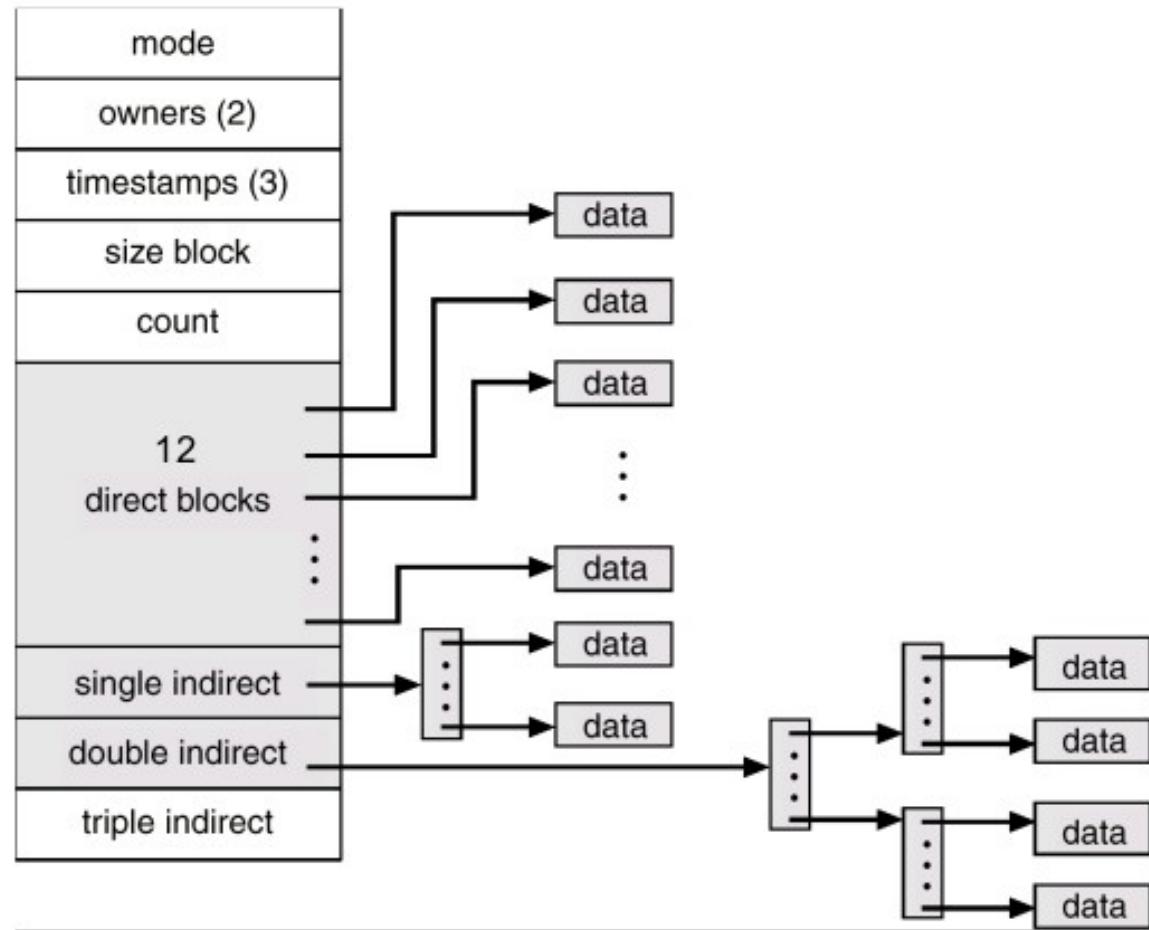


# Sistema de Arquivos

- Indexação multi-nível
  - Custo de acesso é (alto) e igual entre arquivos pequenos e grandes
- I-Node
  - Primeiros blocos → indexação direta

# I-Node

- 12 end. diretos
- 3 end. índices
  - Single
  - Double
  - Triple

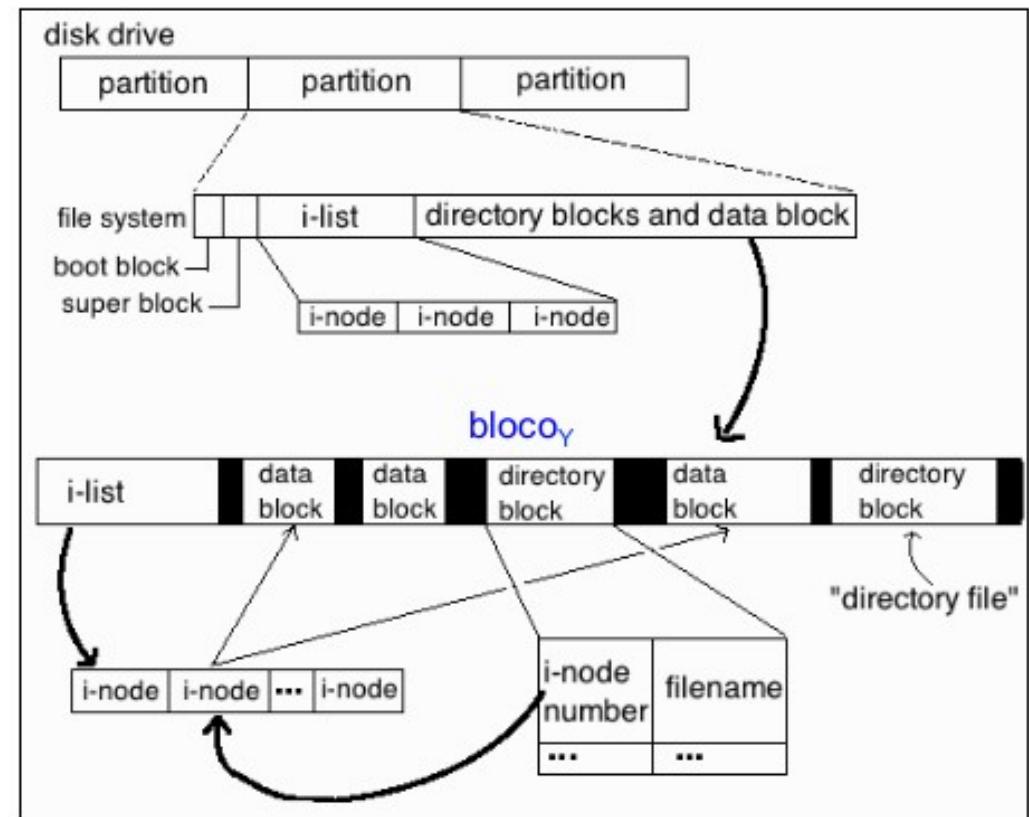


# Sistema de Arquivos - Ext2

- I-Node representa arquivos e diretórios
  - Cada entrada de diretório contém
    - um nome (string variável)
    - Um número (endereço i-node) → índice para uma entrada do vetor de i-nodes da partição correspondente

# Acessando Arquivos

- Arquivo X do dir Y
  - I-node bloco<sub>Y</sub>
  - Acessar bloco<sub>Y</sub>
  - I-node de X
  - Acessar blocos de X



# Acessando Arquivos

- Exemplo

## resolução de /usr/ast/src

Root directory	
1	.
1	..
4	bin
7	dev
14	lib
9	etc
6	usr
8	tmp

Looking up  
usr yields  
i-node 6

I-node 6  
is for /usr  
Mode  
size  
times  
132

I-node 6  
says that  
`/usr` is in  
block 132

6	•
1	••
19	dick
30	erik
51	jim
26	ast
45	bal

/usr/ast  
is i-node  
26

I-node 26  
is for  
/usr/ast

I-node 26  
says that  
/usr/ast is in  
block 406

26	•
6	••
64	grants
92	books
60	mbox
81	minix
17	src

/usr/ast/mbox  
is i-node  
60