

# ***Curso de Engenharia de Computação*** ***Sistemas Operacionais***

INSTITUTO MAUÁ DE TECNOLOGIA



## **Sistemas de Arquivo – Parte IV**



Slides da disciplina Sistemas Operacionais  
Curso de Engenharia de Computação  
Instituto Mauá de Tecnologia – Escola de Engenharia Mauá  
Prof. Marco Antonio Furlan de Souza

## ■ Tamanho de bloco

- **Escrever os bytes** de um **arquivo de modo contíguo** no disco possui **desvantagens**;
- Então um **arquivo** é “**fatiado**” em **blocos de tamanho fixo** – **decisão**: qual deve ser o **tamanho** desses **blocos**?
  - **Blocos grandes desperdiçam espaço** em disco (principalmente se houver uma grande quantidade de arquivos pequenos);
  - **Blocos pequenos** levam a uma **grande quantidade** de **operações de rotação e posicionamento** no disco – **atrasam as operações de disco**.

# Implementação de sistema de arquivos

## ■ Tamanho de bloco

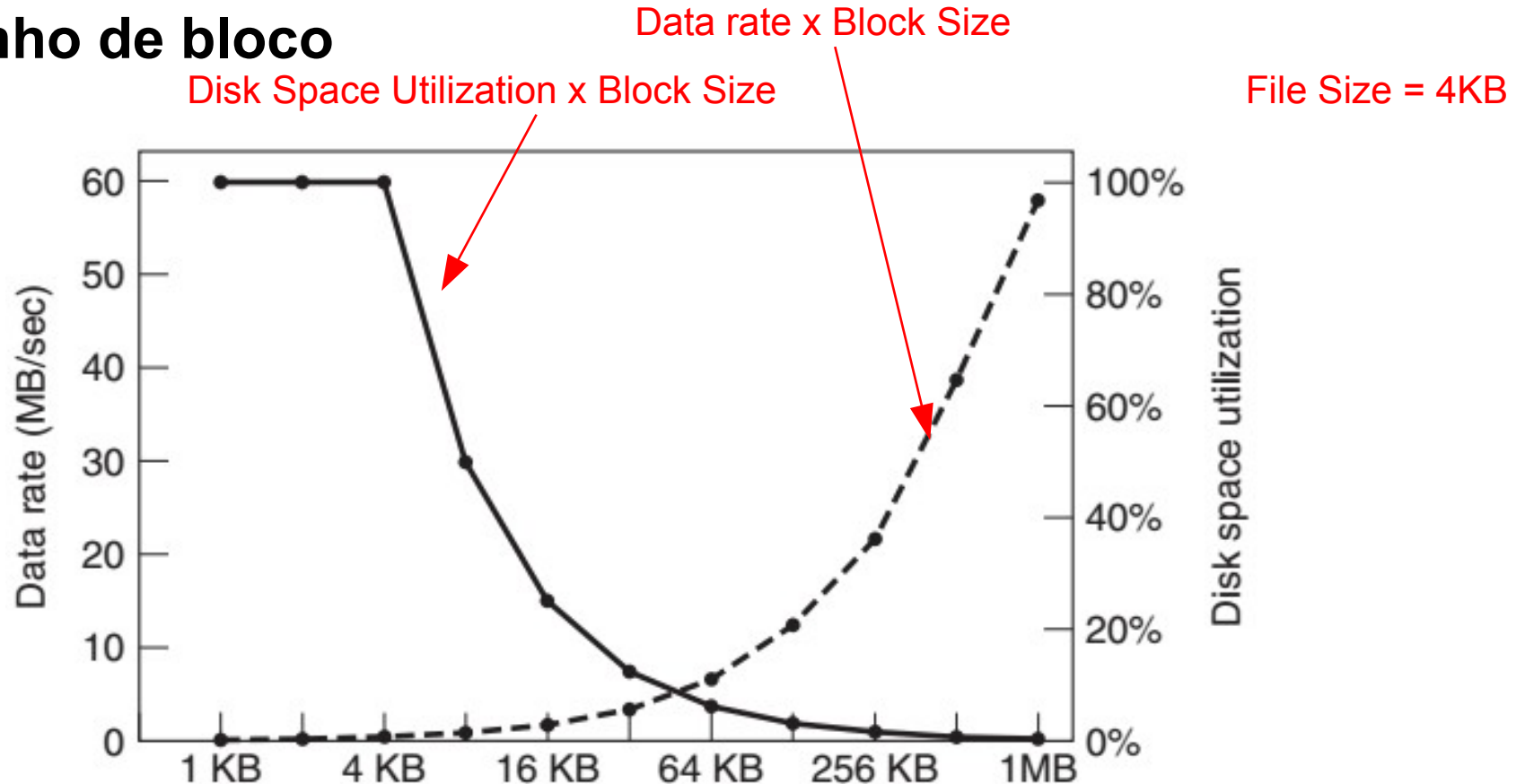
- Porcentagem de arquivos menores que um tamanho dado

Length	VU 1984	VU 2005	Web
1	1.79	1.38	6.67
2	1.88	1.53	7.67
4	2.01	1.65	8.33
8	2.31	1.80	11.30
16	3.32	2.15	11.46
32	5.13	3.15	12.33
64	8.71	4.98	26.10
128	14.73	8.03	28.49
256	23.09	13.29	32.10
512	34.44	20.62	39.94
1 KB	48.05	30.91	47.82
2 KB	60.87	46.09	59.44
4 KB	75.31	59.13	70.64
8 KB	84.97	69.96	79.69

Length	VU 1984	VU 2005	Web
16 KB	92.53	78.92	86.79
32 KB	97.21	85.87	91.65
64 KB	99.18	90.84	94.80
128 KB	99.84	93.73	96.93
256 KB	99.96	96.12	98.48
512 KB	100.00	97.73	98.99
1 MB	100.00	98.87	99.62
2 MB	100.00	99.44	99.80
4 MB	100.00	99.71	99.87
8 MB	100.00	99.86	99.94
16 MB	100.00	99.94	99.97
32 MB	100.00	99.97	99.99
64 MB	100.00	99.99	99.99
128 MB	100.00	99.99	100.00

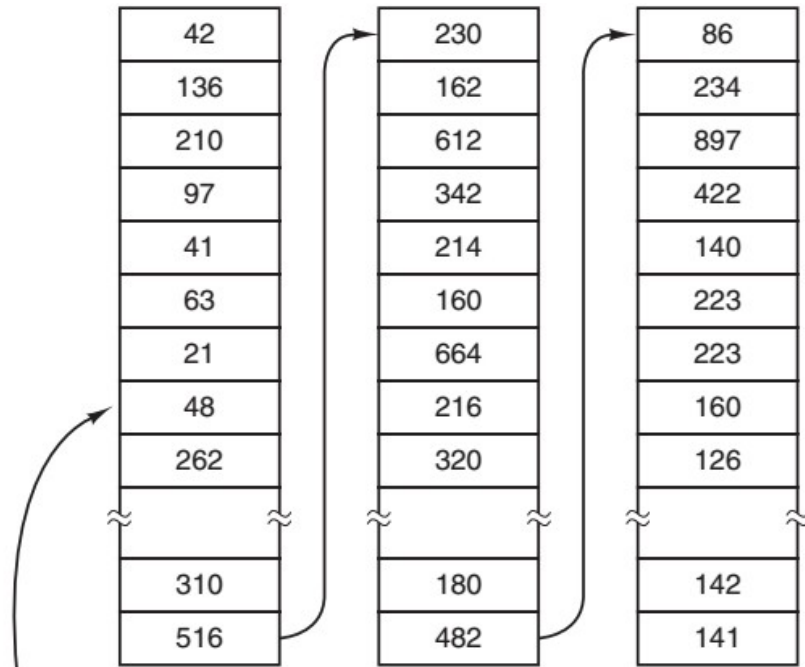
# Implementação de sistema de arquivos

- Tamanho de bloco



## ■ Gerenciando blocos livres

- **Método 1: Lista ligada de blocos de disco** – cada slot de bloco armazena o número de bloco de disco livre.



A 1-KB disk block can hold 256  
32-bit disk block numbers

Por exemplo, com tamanho de bloco de 1KB e 32 bits para identificar um bloco no disco, é necessário  $1024 \times 8 / 32 = 256$  slots (mas um é utilizado para encadear a lista)

Basta um bloco na memória: os demais podem ser lidos/escritos em demanda do disco.

## ■ Gerenciando blocos livres

- **Método 2: Utilizar bitmap** – um disco com n blocos requer um bitmap com n bits – blocos livres são marcados com 1 no bitmap.

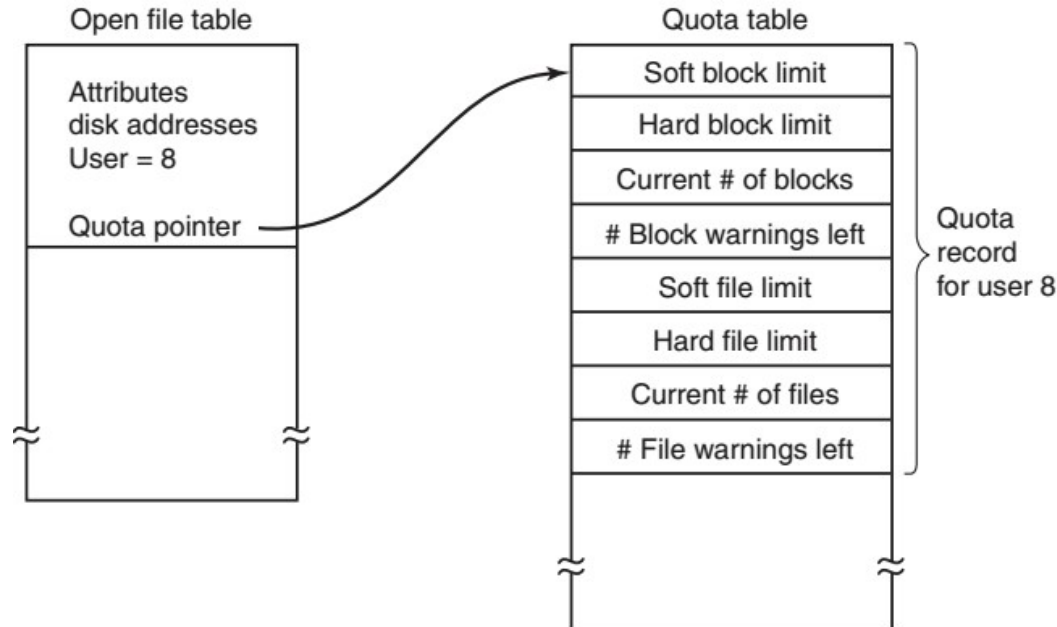
1001101101101100
0110110111110111
1010110110110110
0110110110111011
1110111011101111
1101101010001111
0000111011010111
1011101101101111
1100100011101111
≈
0111011101110111
1101111101110111

Por exemplo, um disco de 1TB  
é necessário um bitmap de  
1bilhão de bits – requer  
130000 blocos de 1KB para  
armazenar.

# Implementação de sistema de arquivos

## ■ Quotas de disco

- Utilizado pelo administrador para **limitar** o **tamanho** de **bloco** e de **arquivo** dos **usuários** – requer informações adicionais associadas a um arquivo aberto:





# *Implementação de sistema de arquivos*

- **Backups do sistema de arquivos**
  - **Backups** são utilizados para:
    - Recuperar de **desastres**;
    - Recuperar de **estupidezas...**
  - **Estratégias:**
    - Usar o conceito de “**lixeira**”;
    - **Executar** uma **cópia** do **sistema**.



- **Backups do sistema de arquivos**
  - **Problemas com cópias de backup**
    - É um **processo lento** – escolher que arquivos devem ser feitos backup e que tipo de mídia (segurança);
    - **Backup incremental**: é mais rápido – cria-se um backup inicial, grande e então se adicionam e se atualizam apenas novos e modificados arquivos – mas é mais complicado de gerir;
    - **Uso de compressão de dados**: economiza espaço, mas um erro na compressão pode levar à perda de uma parte significativa dos dados;
    - **É complicado fazer o backup de um sistema de arquivos ativo** – o que fazer com arquivos que são modificados durante o backup?
    - **Proteção**: como proteger o backup?

- **Backups do sistema de arquivos**
  - **Abordagens de gravação (*dump*)**
    - **Físico:** gravam-se todos os blocos do disco a partir do bloco zero, em ordem, em um disco de saída. É rápido, mas não deixa escolher diretórios ou arquivos e, além disso, copia blocos vazios e também blocos ruins (se houver);
    - **Lógico:** inicia-se em um ou mais diretórios e então procede-se à cópia de modo recursivo. No UNIX, este processo copia todos os caminhos inteiros de arquivos modificados ou adicionados e utiliza um esquema de bitmaps para marcar essas modificações.

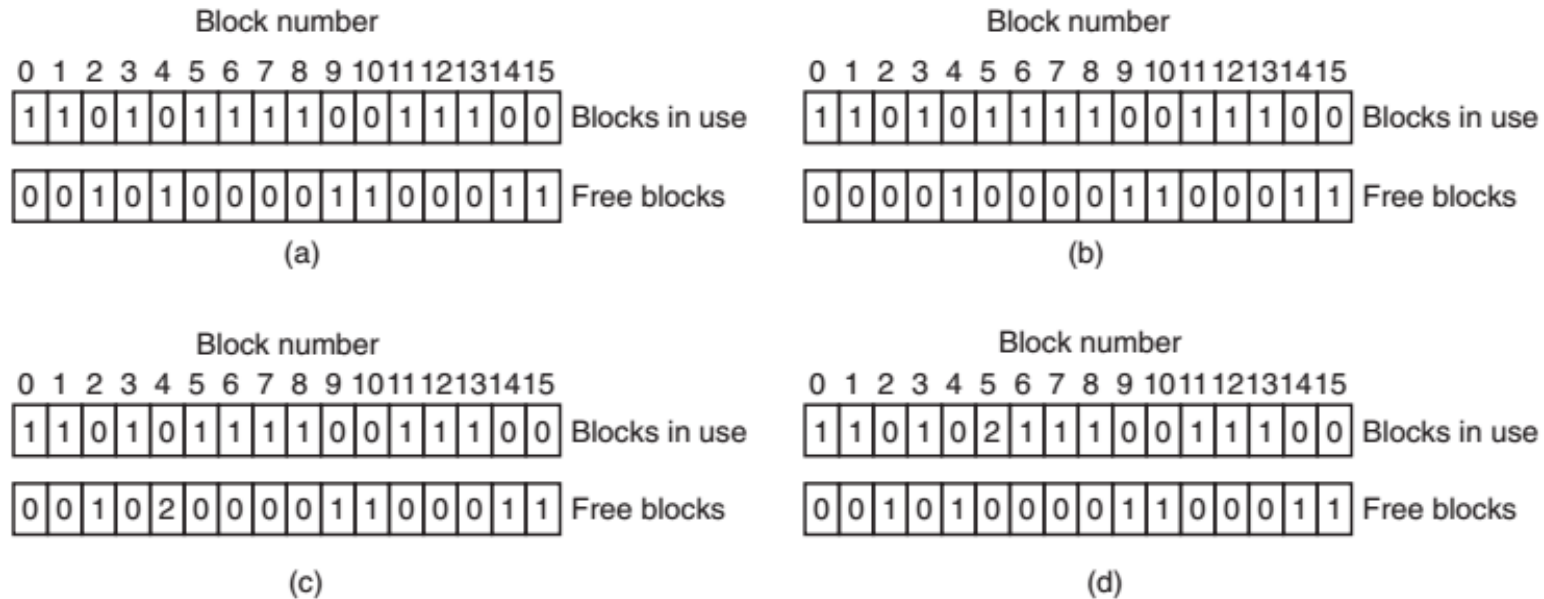
## ■ Consistência do sistema de arquivos

- Para lidar com **inconsistências** (blocos que não foram gravados e estão “soltos” no sistema após um crash), existem **programas** que tentam **verificar** e **possivelmente consertar** esses problemas;
- **Exemplo:** o programa **fsck** do UNIX/Linux – busca por **inconsistências** em **arquivos** e **blocos** – utiliza **duas tabelas** – em uma ele **marca**, após **seguir** os **i-nodes** de um **arquivo**, qual **bloco** estava em **uso**; na **outra** ele **segue** uma **lista ligada** ou **bitmap** de **blocos** livres. Em um **sistema** consistente, o **número 1** **aparecerá** em uma **tabela** ou em **outra**.

# Implementação de sistema de arquivos

## ■ Consistência do sistema de arquivos

### – Exemplo



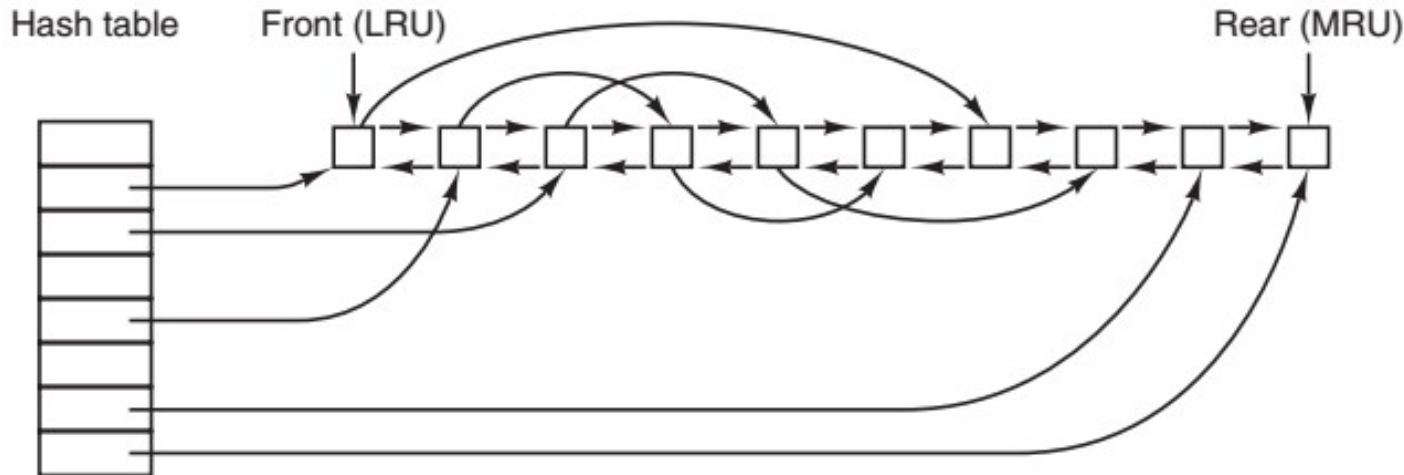
**Figure 4-27.** File-system states. (a) Consistent. (b) Missing block. (c) Duplicate block in free list. (d) Duplicate data block.

# Implementação de sistema de arquivos

- Desempenho do sistema de arquivos

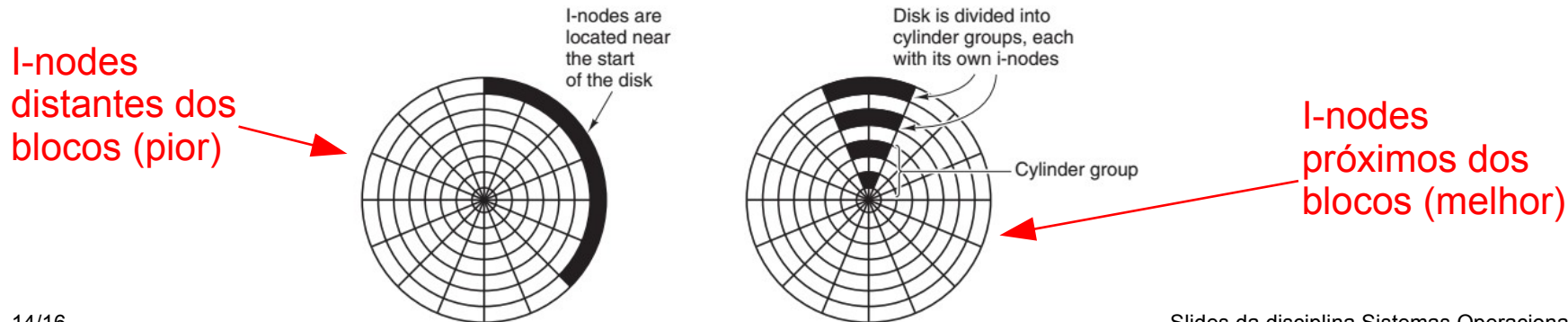
- Caching

- Consiste em manter uma **lista** de **blocos** de arquivo **mais** utilizados em um **cache**;
    - A utilização de uma **tabela** de **hash** **acelera** a **localização** no cache:



# Implementação de sistema de arquivos

- **Desempenho do sistema de arquivos**
  - **Leitura de bloco adiantada**
    - Consiste em **adiantar blocos** que poderão ser necessários e já **armazená-los** no **cache**;
  - **Redução do movimento do braço do disco**
    - Tentar **armazenar blocos** que poderão ser **lidos** na **sequência** no mesmo **cilindro** – no caso do UNIX/Linux, armazenar i-nodes nos cilindros do meio é melhor:



- **Desempenho do sistema de arquivos**
  - **Desfragmentação de discos**
    - Após a **utilização** do **sistema de arquivos**, é comum aparecer **espaços resultantes** de sua **manipulação** que são **inúteis** para **alocar arquivos** de tamanho maior, resultando no **espalhamento** do **arquivo** e **consequentemente** uma **perda de desempenho**;
    - A **desfragmentação** consiste em **reunir** esses **espaços** (movendo blocos de arquivos) criando **regiões contíguas** de blocos.



# ***Referências bibliográficas***

TANENBAUM, Andrew S. **Sistemas operacionais modernos**. 3. ed.  
São Paulo: Pearson, 2013. 653 p.