

Sistemas de Arquivos

Diretórios

Gerenciamento de Espaço em Disco

Implementação de Diretórios (1)

- Contém informações que permitem acessar os arquivos
 - As entradas do diretório fornecem informações para encontrar os blocos de discos
- Possui vários campos, uma por arquivo:
 - nome
 - tipo; tamanho
 - proprietário; proteção
 - data de criação; data da última modificação
 - **lista de blocos usados**

Implementação de Diretórios (2)

games	attributes
mail	attributes
news	attributes
work	attributes

(a)

MS-DOS/WINDOWS

games	
mail	
news	
work	

(b)

UNIX/LINUX

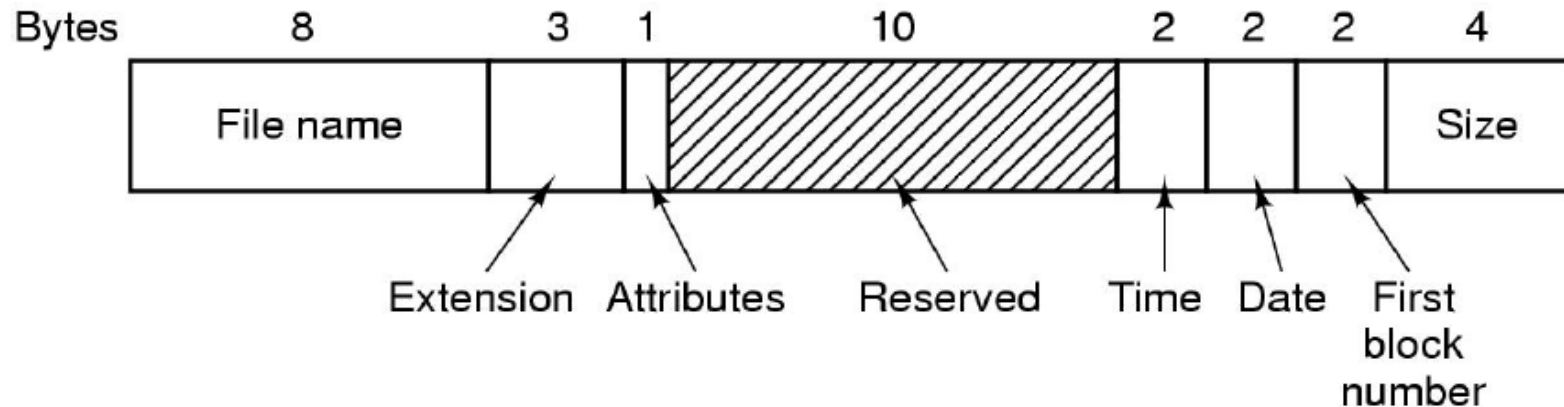


Data structure
containing the
attributes

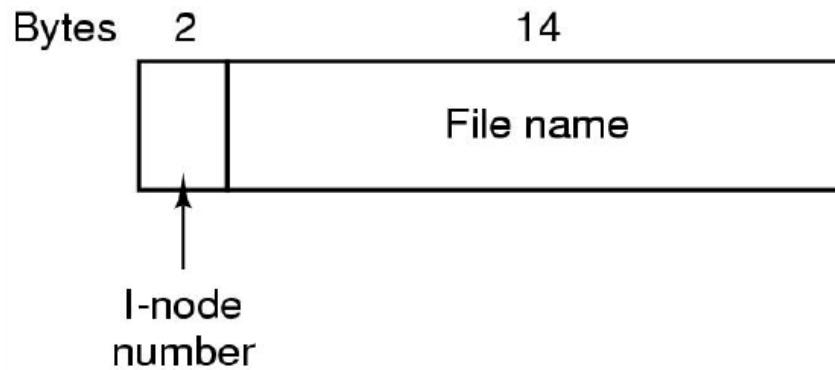
- (a) Diretório simples com
 - Entradas de dimensão fixa
 - Endereços de disco e atributos na entrada de diretório
- (b) Diretório onde cada entrada apenas refere um i-node (que mantém os atributos)

Implementação de Diretórios (3)

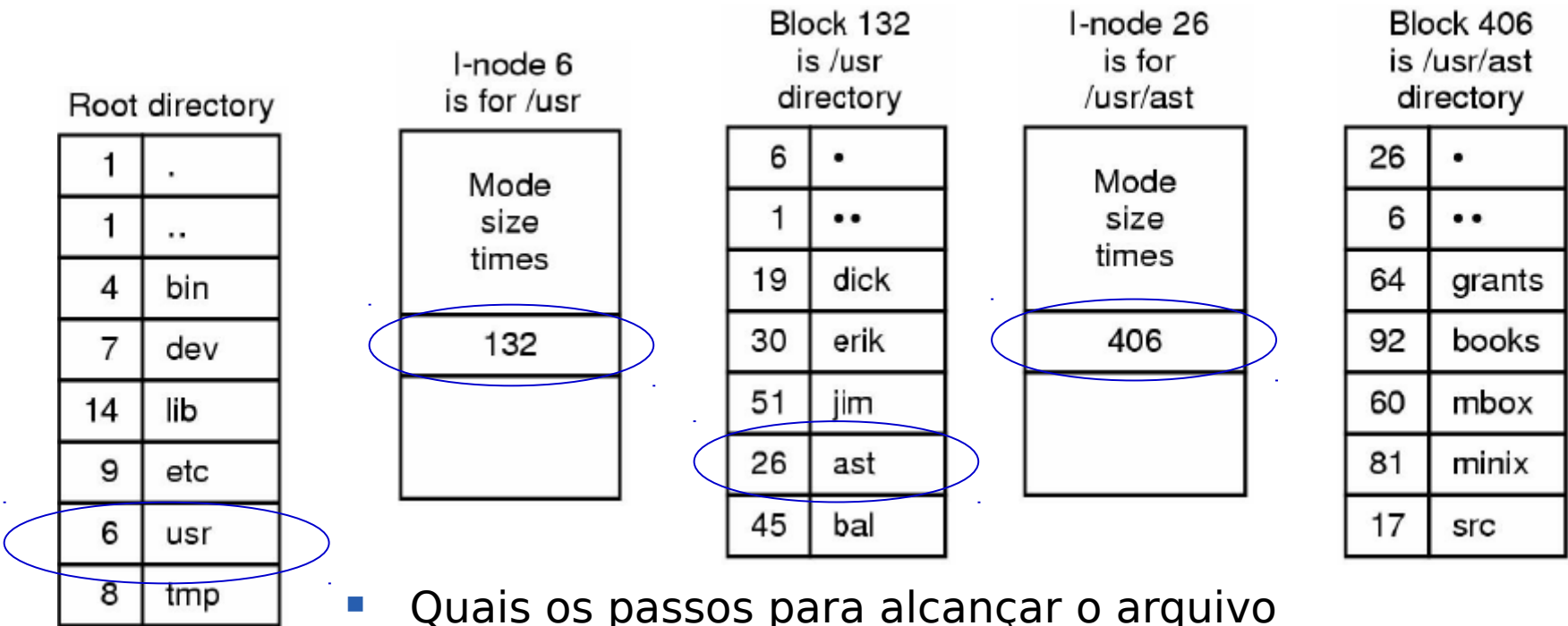
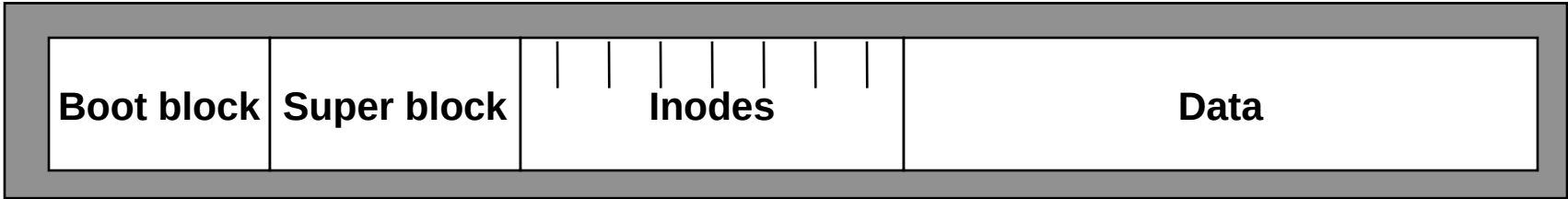
- Entrada de diretório no DOS



- No Unix (eg. System V)



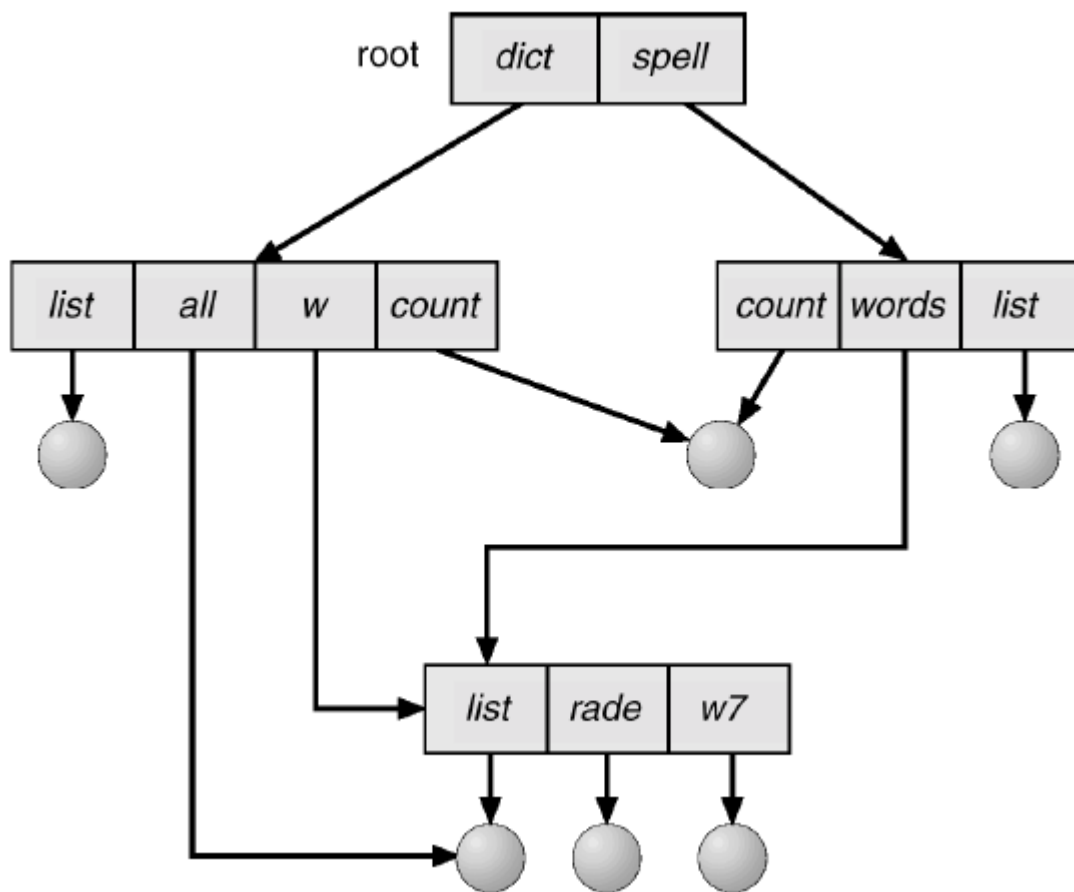
Implementação de Diretórios (5)



■ Quais os passos para alcançar o arquivo `/usr/ast/mbox`?

Arquivos Compartilhados ⁽¹⁾

- Hierarquia de diretórios: Grafo acíclico orientado

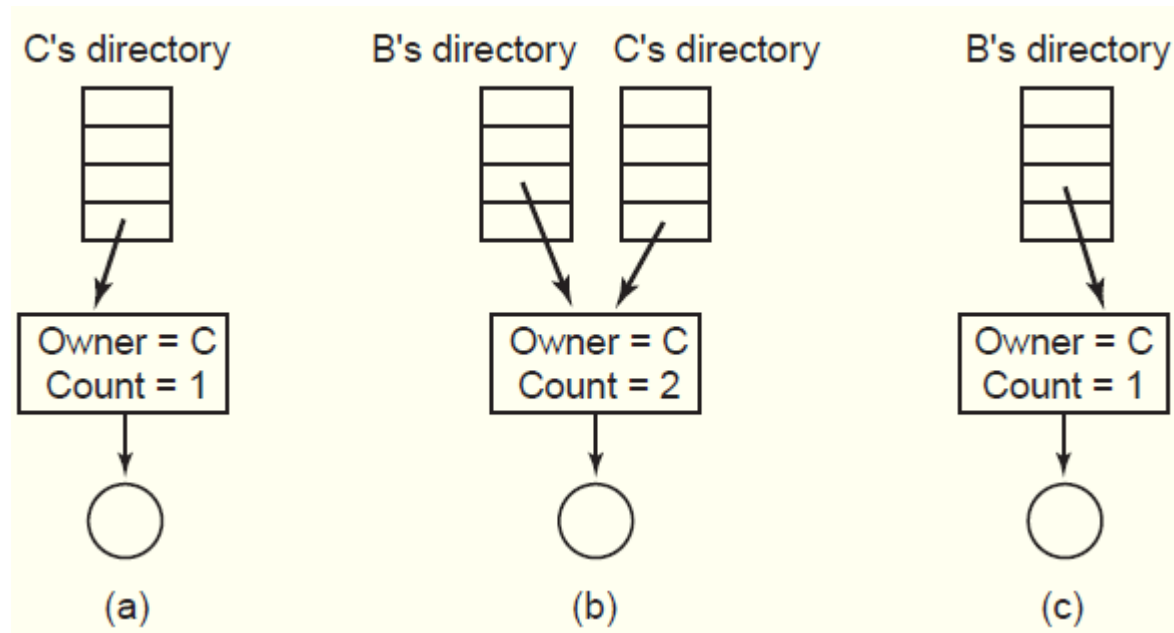


Arquivos Compartilhados (2)

- O diretório contém a lista (endereços) dos blocos que pertencem ao arquivo
 - é feita uma cópia dos endereços dos blocos para o diretório do arquivo “link”
 - Problema: não existe compartilhamento,
 - mudanças em uma versão (e.g. *append* no fim do arquivo) não são vistas em outra
- Soluções
 - 1a. Solução: os blocos não fazem parte do diretório, mas sim de estruturas de dados associadas aos descritores. O diretório aponta para essa estrutura de dados (UNIX)
 - 2a. Solução: “link” simbólico - o diretório contém o nome do arquivo “linkado”

Arquivos Compartilhados (3)

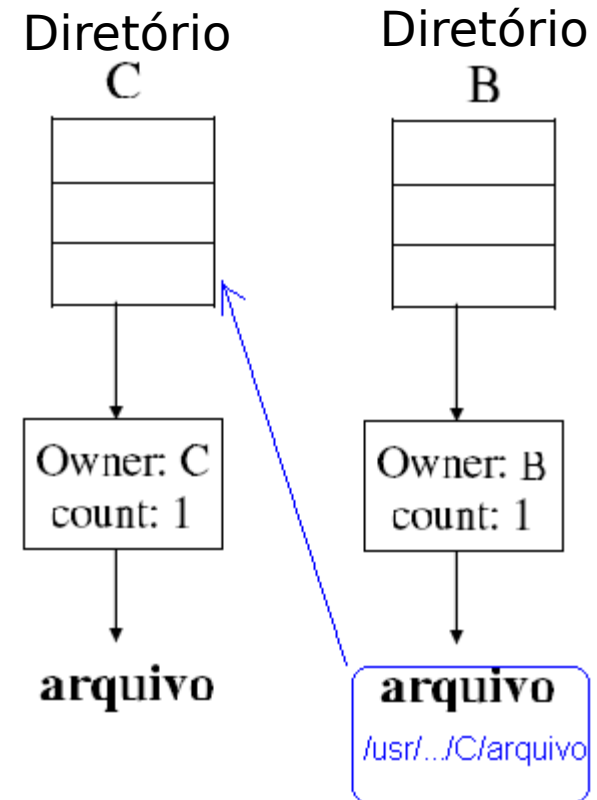
- 1a. Solução



Arquivos Compartilhados (3)

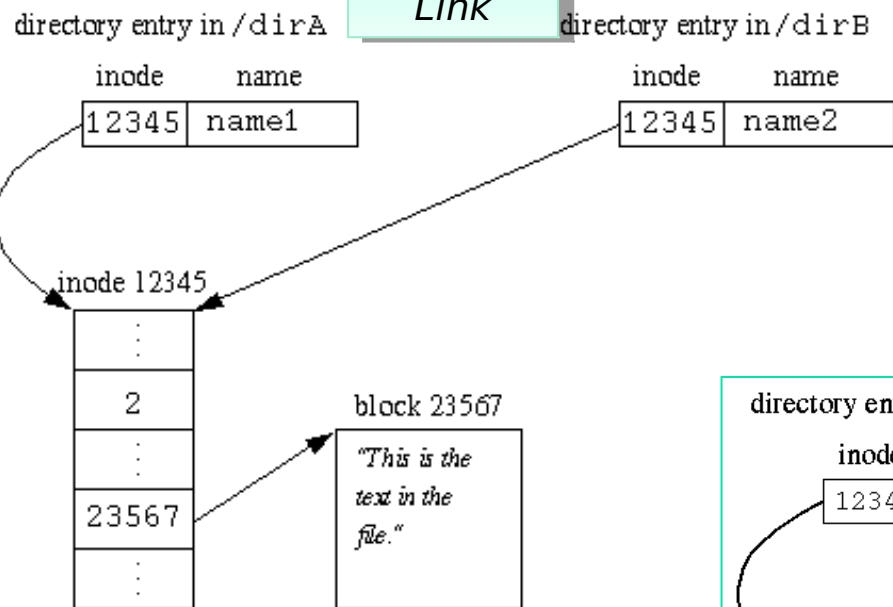
■ 2a. Solução

- não existe o problema de deleção do arquivo por parte do proprietário
- Problema: número de acessos a disco pode ser elevado
- Vantagem: link de arquivos em máquinas diferentes

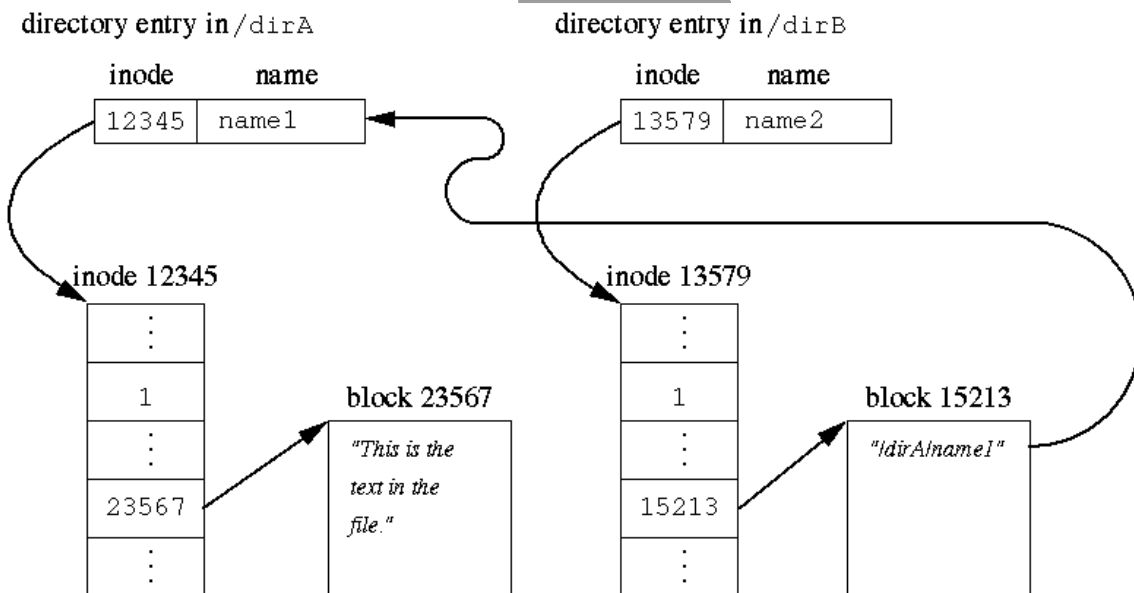


Arquivos Compartilhados (4)

Hard Link



Soft Link



Criando Hard Links (SVC)

```
#include <sys/unistd.h>

int link (const char *path1, const char *path2);
    // Cria um hard link (path2 -> path1)
int unlink (const char *path1, const char *path2);
    // Apaga um hard link
```

■ Exemplo: criando um hard link

```
#include <stdio.h>
#include <sys/stat.h>
...
    if (link("/dirA/name1", "/dirB/name2") == -1)
        perror("Failed to make a new link in /dirB");
...
```

```
ln /dirA/name1 /dirB/name2
```

Criando Soft Links (SVC)

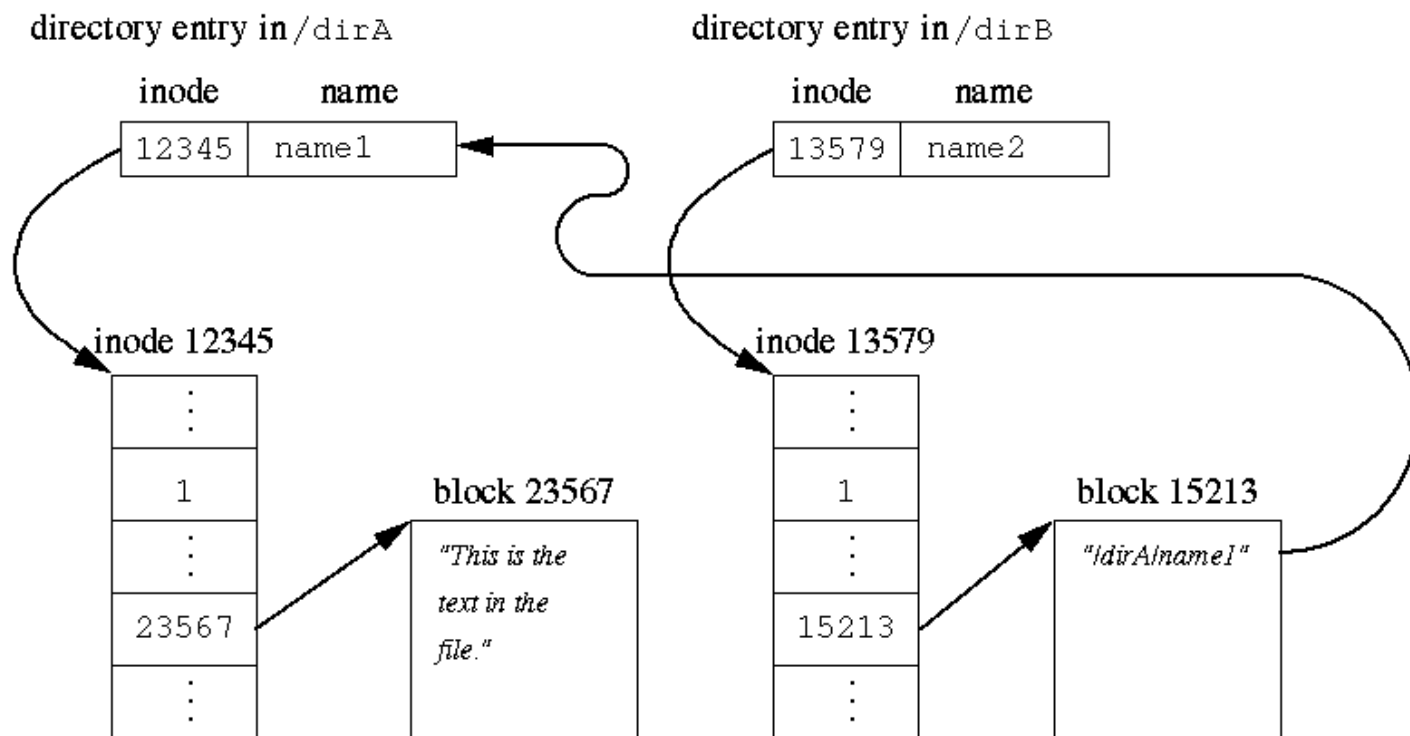
- SVC p/ criação de link simbólicos

```
#include <sys/unistd.h>

int symlink (const char *path1, const char *path2);
// Cria um link simbólico (path2 -> path1)
```

~

```
ln -s path1 path2
```



Gerenciamento de Espaço em Disco (1)

- Tamanho de Bloco

- Bloco Grande

- Menos acessos a disco
 - Aumenta fragmentação interna

- Bloco Pequeno

- Diminui a fragmentação interna
 - Arquivo contendo muitos blocos => acesso mais lento

- Tempo para se ler um bloco

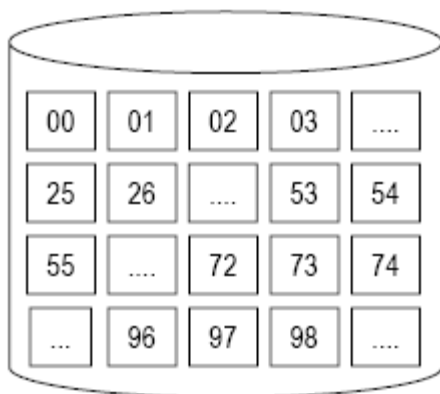
Tmp médio de seek + latência rotacional + tempo de leitura

- Exemplo:

- Considere um disco com 131.072 bytes por trilha, tmp de rotação= 8,33 ms
tmp médio de seek = 10 ms. Tmp p/ ler um bloco de k bytes?
 - ler um bloco de k bytes = $10 + 4,16 + (k/131072) \times 8,33$

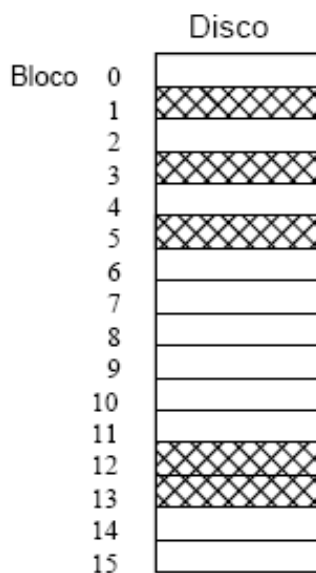
Gerenciamento de Espaço em Disco (2)

- Gerenciamento do Espaço Livre
 - Necessário manter a informação de blocos livres e ocupados
 - Métodos Básicos
 - Mapa de bits
 - Lista de blocos livres
 - Ambos os métodos consideram que os blocos são numerados seqüencialmente



Gerenciamento de Espaço em Disco (3)

- Gerenciamento do Espaço Livre (cont.)
 - Mapa de bits
 - Forma simples de gerenciar o espaço em disco
 - Um disco c/ n blocos requer um bitmap de n bits



Mapa de Bits.

0	0	1	0	1	0	1	0
0	0	1	1	0	0	0	0

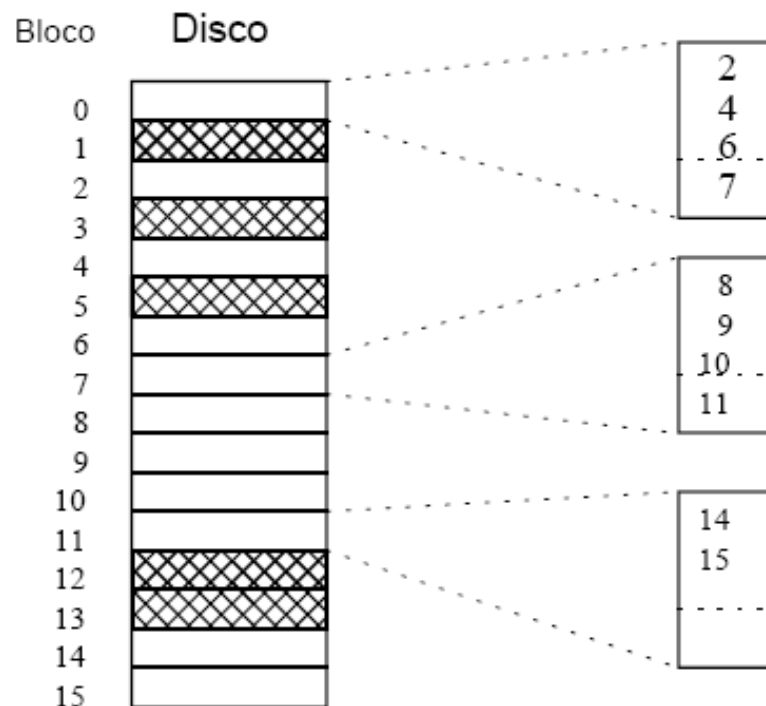
$$tamanho_bit_map = \frac{Capacidade_disco(bytes)}{8 \times tamanho_bloco(bytes)}$$

Gerenciamento de Espaço em Disco (4)

■ Gerenciamento do Espaço Livre (cont.)

■ Lista encadeada de blocos livres

- A lista é mantida no próprio disco
- Problema: tamanho da lista
- Paliativo: a medida que o espaço em disco é ocupado, a lista diminui, liberando blocos de tamanho
- Alternativa: manter uma lista de “áreas livres” ao invés de blocos



Gerenciamento de Espaço em Disco (4)

■ Gerenciamento do Espaço Livre (cont.)

■ Lista de blocos livres

- Quantos blocos precisamos para armazenar a lista de blocos livres?
- Exemplo:
 - Disco de 40 M com Blocos de 1k
 - 2 bytes para identificar bloco livre

Cada bloco pode conter 512 entradas

Disco de 40 M => dividido em 40 k blocos

$$\text{nº de blocos} = \frac{40k}{512} = 80 \text{ blocos}$$