

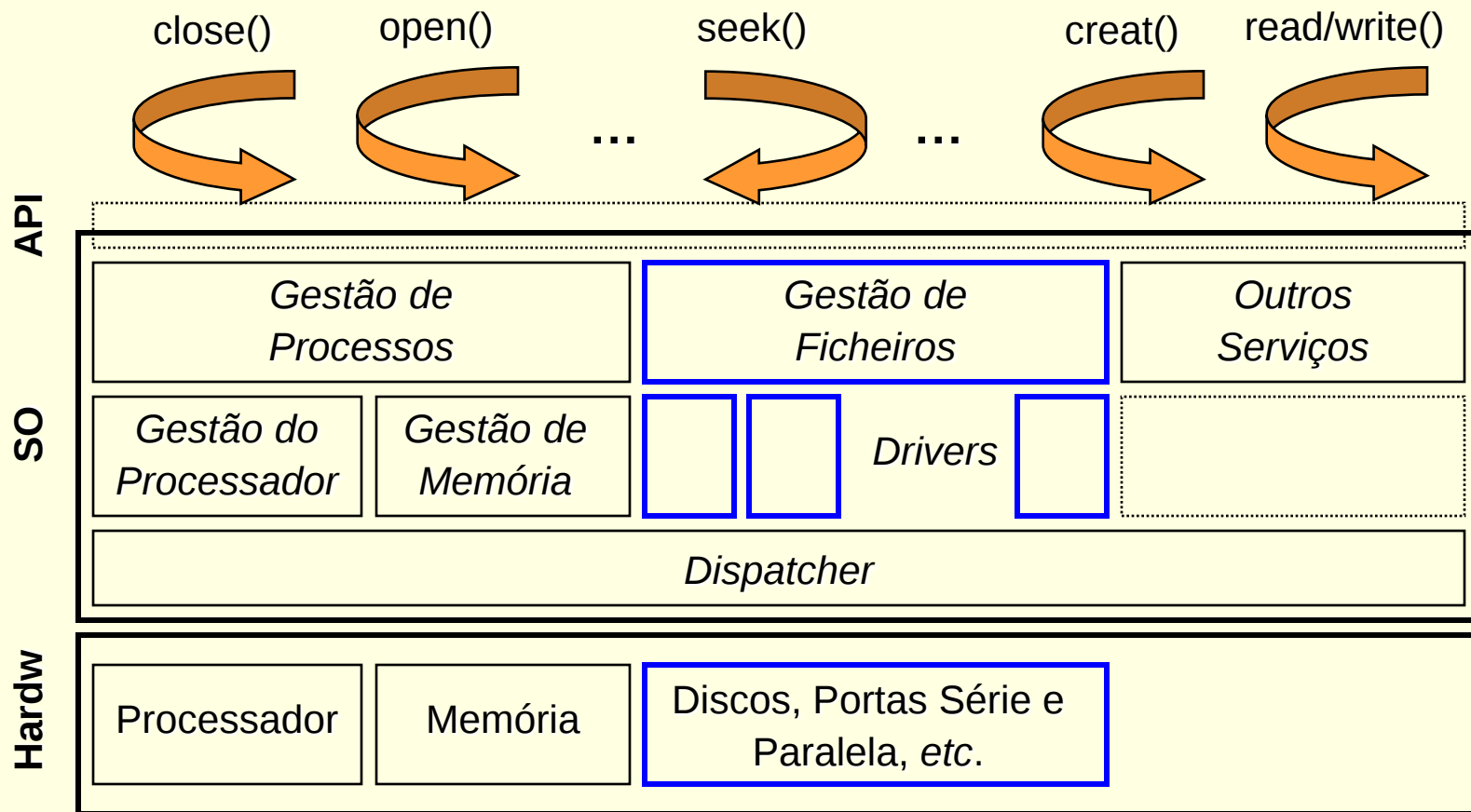
Fundamentos de Sistemas de Operação

Unix Windows NT Network MacOS DOS/VS Vax/VMS
Linux Solaris HP/UX AIX Mach Chorus

Parte II: Gestão de Ficheiros

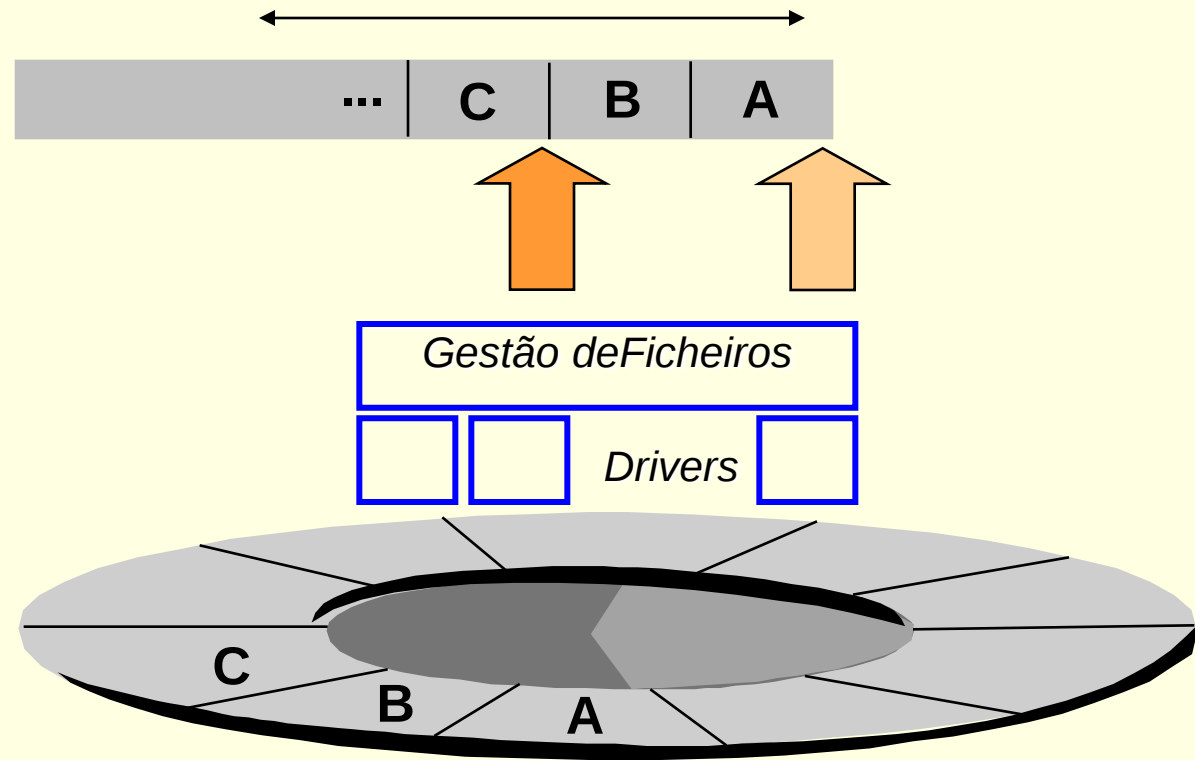
- a) Sistema de Ficheiros em Disco –
Algumas questões...

O Sistema de Ficheiros



Sistemas de Ficheiros: Questões I

- Como implementar sobre o disco o conceito de ficheiro?



Sistemas de Ficheiros: Questões II

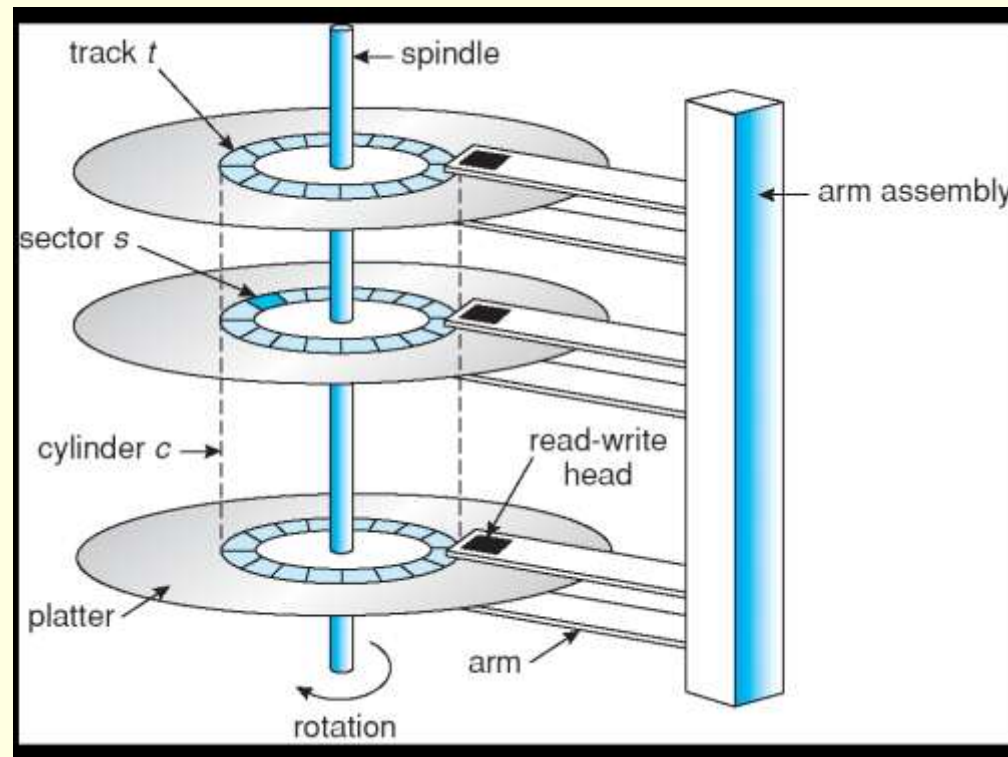
- Como identificar os ficheiros? Que regra para os nomes?
- Como saber onde começa e acaba um ficheiro?
- Como saber que blocos pertencem a um dado ficheiro?
- Dois elementos de dados sequencialmente colocados no ficheiro (lógico) têm de estar em posições contíguas do disco (físico)?

Organização do Disco: Questões

- Como é organizado um disco? E são todos organizados da mesma forma? E, se não são, podem-se esconder as diferenças?
- Como é que o (*software* do) Sistema de Ficheiros (SF) vê o disco? O modelo apresentado pelo controlador e/ou pelo *driver* ao SF é o do verdadeiro disco físico, ou é um modelo abstracto e simplificado?

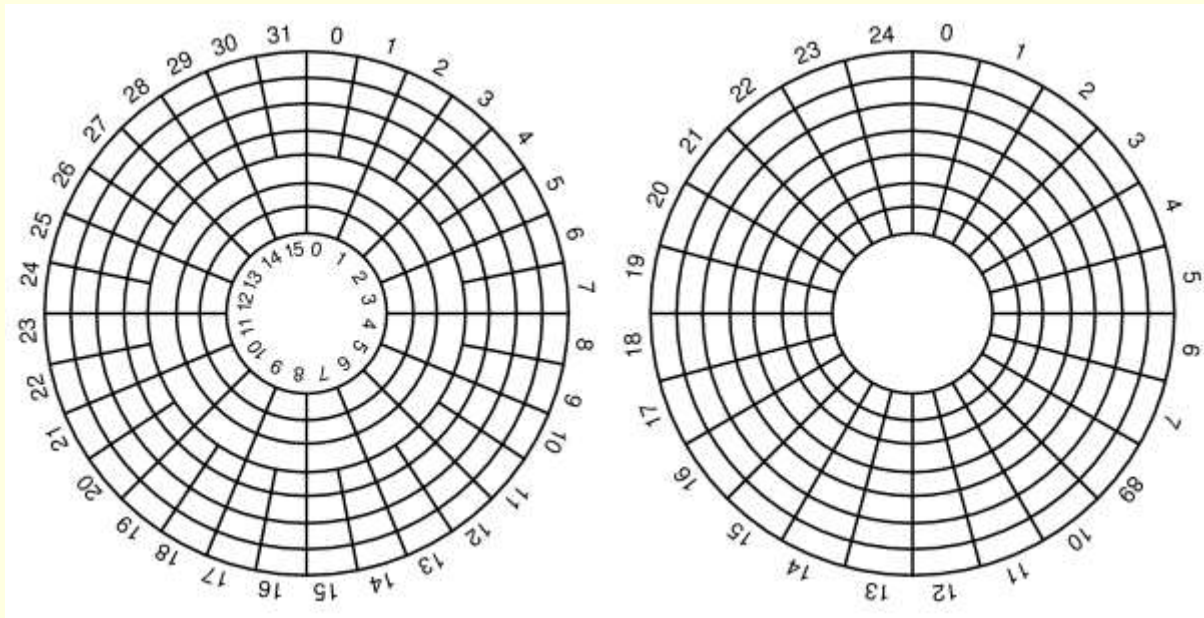
Como é organizado um disco: I

- Múltiplos pratos:



Como é organizado um disco: II

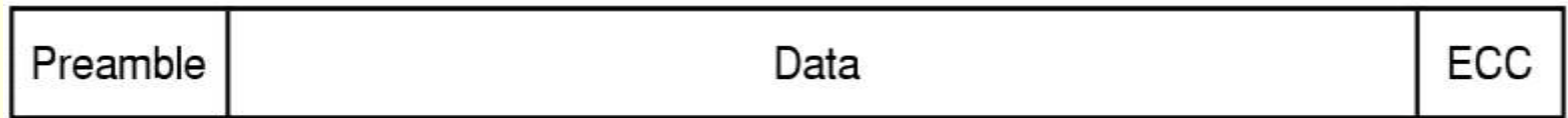
- Geometria real (esq^a) e virtual (dt^a)
 - Mantendo-se a mesma densidade de gravação da periferia para o centro, no centro haverá menos sectores por pista.



Contudo, o controlador pode apresentar uma geometria (virtual) homogénea.

Como é organizado um disco: III

- O formato de um bloco de disco:

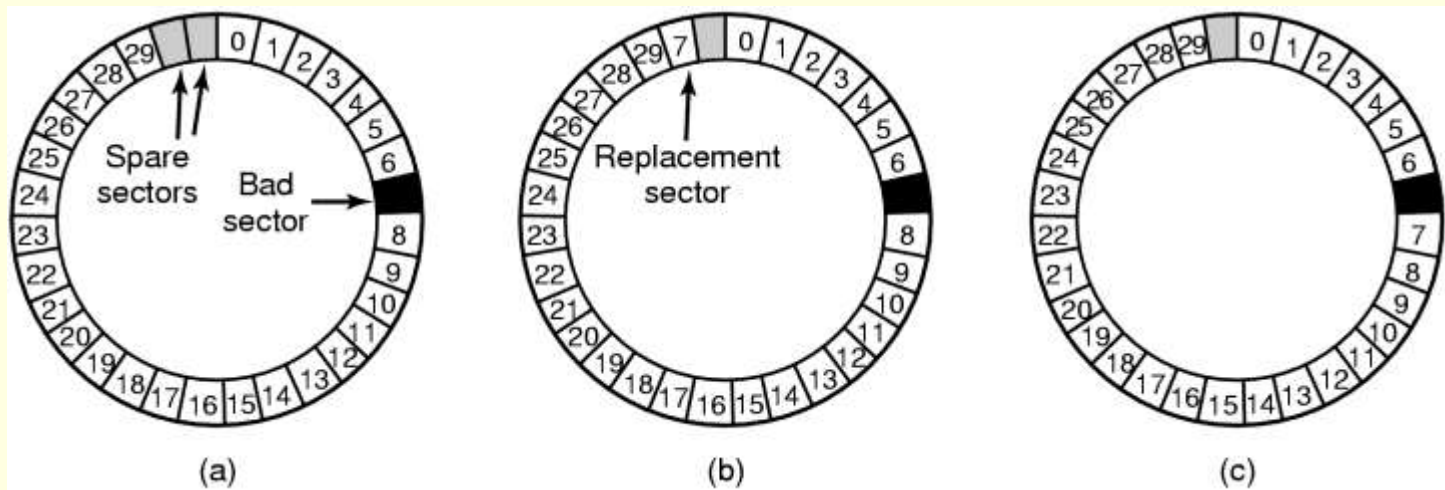


- Preâmbulo: sincronismo e informação de controle.
- Dados: tipicamente 512 bytes.
- ECC: detecção e correcção de erros.

Como é organizado um disco: IV

■ Tratamento de **erros**:

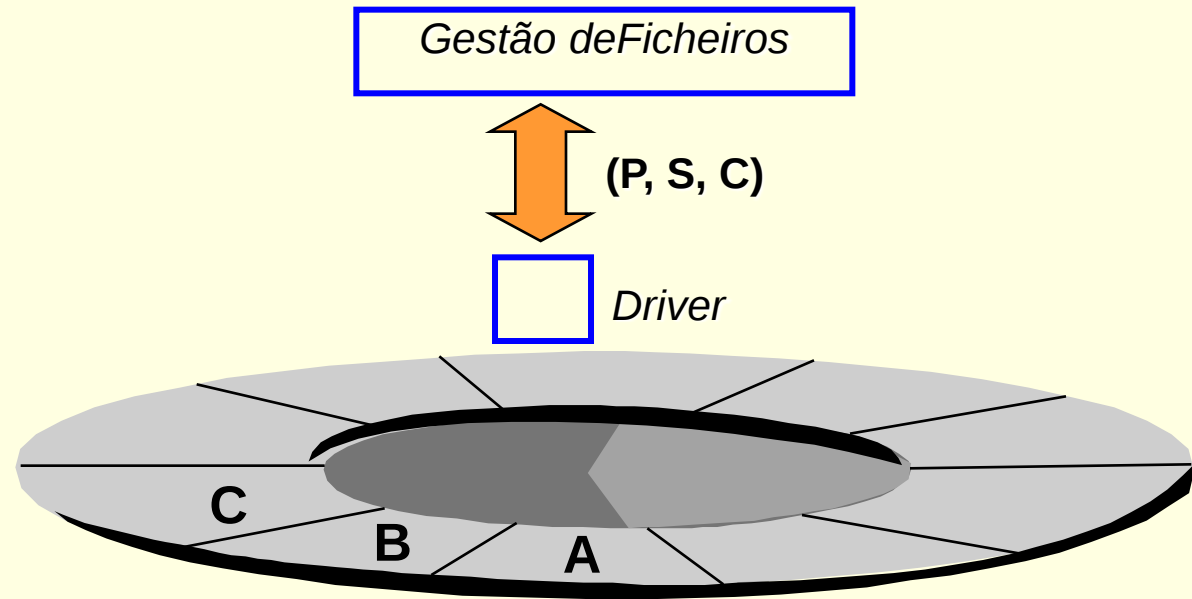
- (a) O bloco 7 estraga-se. (b) é usado um bloco de substituição.



- (c) Alguns controladores poderão ser capazes de re-arranjar a numeração dos blocos para continuar a manter um bom desempenho.

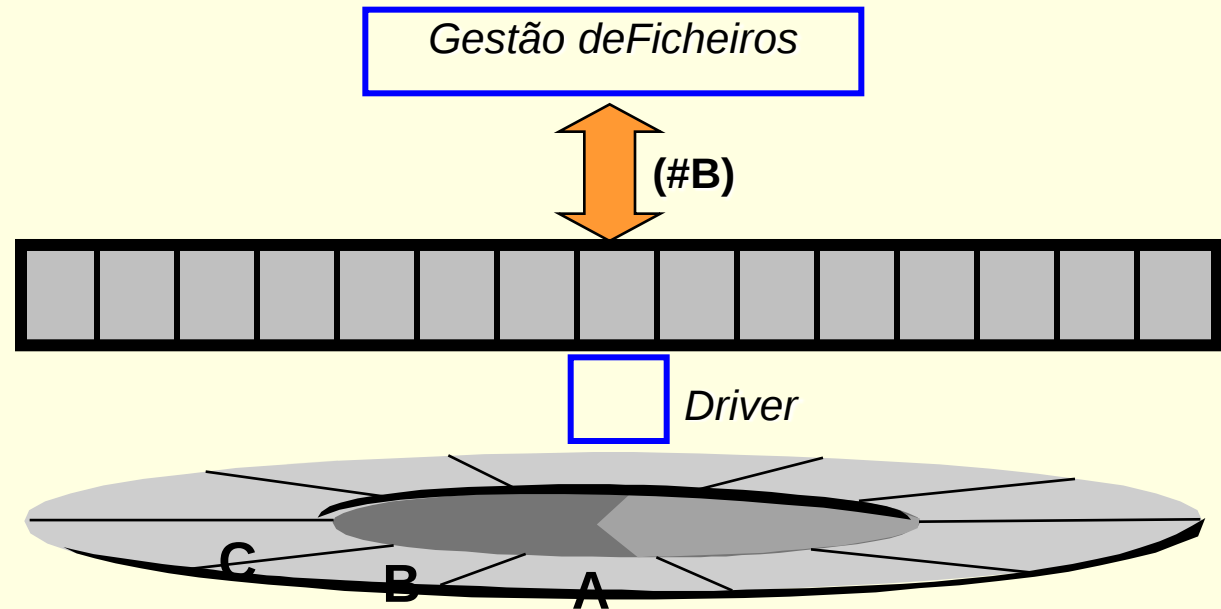
Modelo de Disco apresentado ao SF: I

- O *driver* pode apresentar ao Sistema de Ficheiros um modelo (real ou virtual) em quando o acesso a um bloco é pedido ao driver pelo SF este tem de especificar os números de pista, sector, cabeça onde tal bloco reside...



Modelo de Disco apresentado ao SF: II

- Ou o *driver* pode, por outro lado, apresentar ao Sistema de Ficheiros um modelo de um **vector de blocos** em que o acesso é efectuado especificando apenas o número do bloco, e o driver do disco sabe “onde esse bloco está”...



Fundamentos de Sistemas de Operação

Unix Windows NT Network MacOS DOS/VS Vax/VMS
Linux Solaris HP/UX AIX Mach Chorus

Parte II: Gestão de Ficheiros

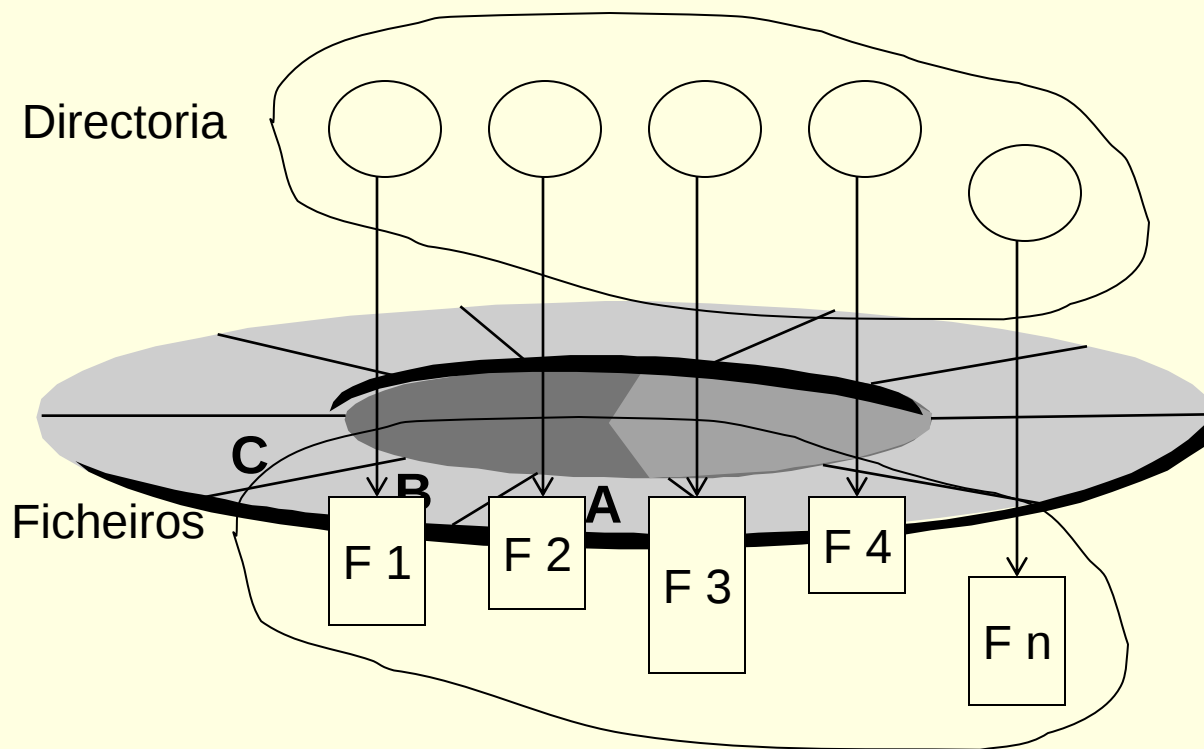
- b) Sistema de Ficheiros em Disco –
O “espaço” de nomes...

Sistemas de Ficheiros: Questões

- Como identificar os ficheiros? Que regra para os nomes?
- Como estruturar o “espaço de nomes”?
 - Um único espaço global “plano” para os nomes?
 - Um espaço independente por utilizador? (e como se partilham ficheiros?)
 - Um espaço hierárquico? (árvore)
 - Um grafo?

Estrutura de directorias

- Colecção de entidades onde é mantida informação sobre todos os ficheiros...



Estrutura de directorias: Objectivos

- Eficiência – localizar rapidamente um ficheiro.
- Designação (*naming*) – adequada às necessidades dos utilizadores:
 - Dois utilizadores podem dar o mesmo nome a ficheiros diferentes.
 - O mesmo ficheiro pode ter vários nomes diferentes.
- Agrupamento – agrupamento lógico de ficheiros de acordo com as suas propriedades: (por ex., programas em Java, jogos, ...)

Directorias: Conteúdo

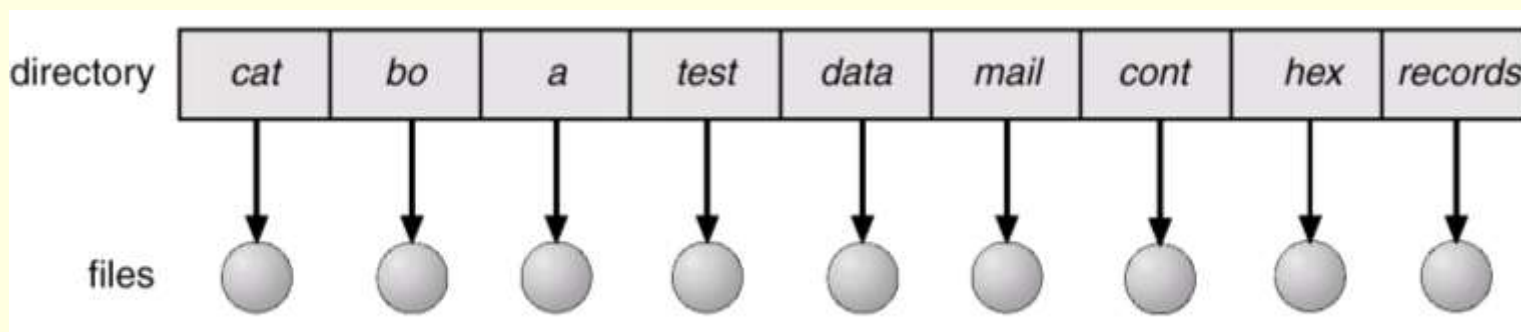
- Para cada “ficheiro”:
 - Nome
 - Tipo
 - Localização
 - Comprimento
 - Data do último acesso (útil para gestão/backup)
 - Data da última actualização (útil para gestão/backup)
 - ID do dono
 - Informação de protecção (confidencialidade)

Directorias: Operações

- Primitivas sobre o “tipo” Directoria:
 - “Localizar” um ficheiro
 - Criar um ficheiro
 - Apagar um ficheiro
 - Listar o conteúdo (obter lista de ficheiros)
 - Mudar o nome de um ficheiro
 - Navegar na estrutura de directorias

Directoria de um nível (única)

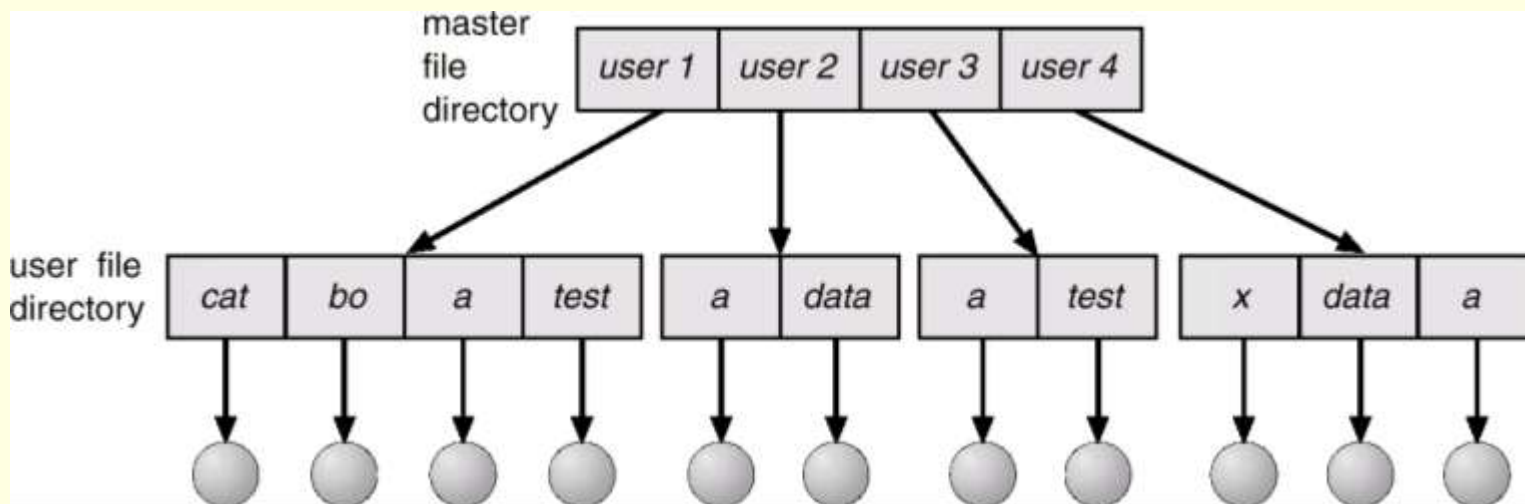
- Uma única Directoria para todos os utilizadores



- Problemas de:
 - Designação: dificuldade de atribuir o nome (colisões)
 - Agrupamento: não é possível

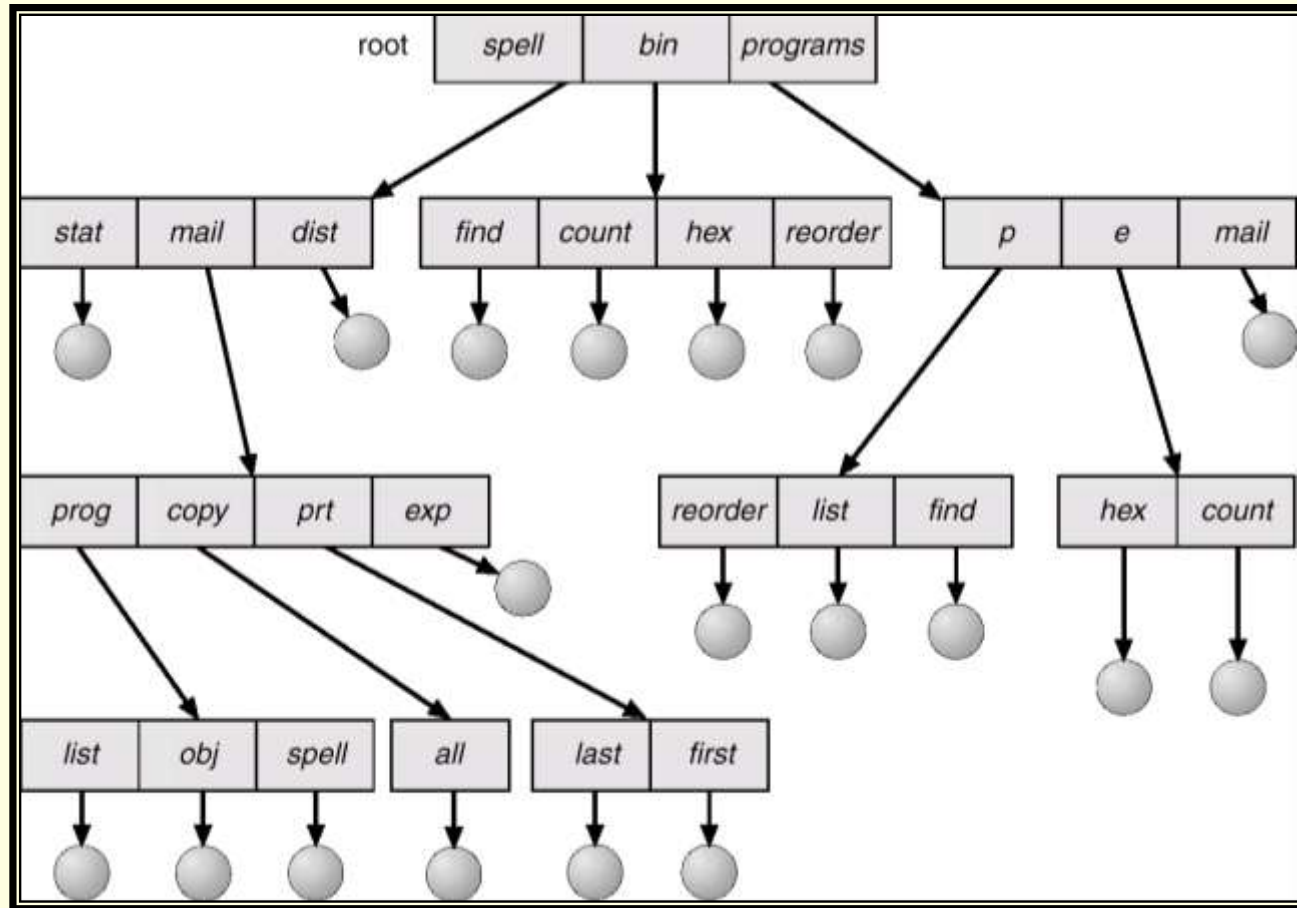
Directoria de dois níveis

- Uma Directoria “privada” para cada utilizador



- Problemas de:
 - Agrupamento (realizado pelo utilizador): não é possível
 - Partilha?

Directorias em árvore: I

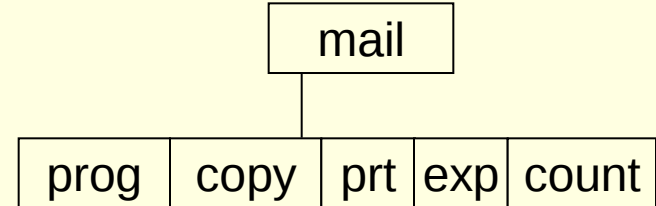


Directorias em árvore: II

- Pesquisa eficiente
- Facilidade de agrupamento
- Directoria corrente (working directory)
 - `cd /home/pal/prog`
 - `.` , `..`
- Nomes absolutos ou relativos
- Criação e destruição de ficheiros e directorias usando nomes relativos ou absolutos
 - `rm <file-name>`
 - `mkdir <dir-name>`

Directorias em árvore: III

- Criação e destruição de ficheiros e directorias: algumas questões, e “cuidados” – ex^o:
 - Se a directoria corrente é /mail, e
 - mkdir count



- ? Que acontece se se fizer
 - rm /mail

Fundamentos de Sistemas de Operação

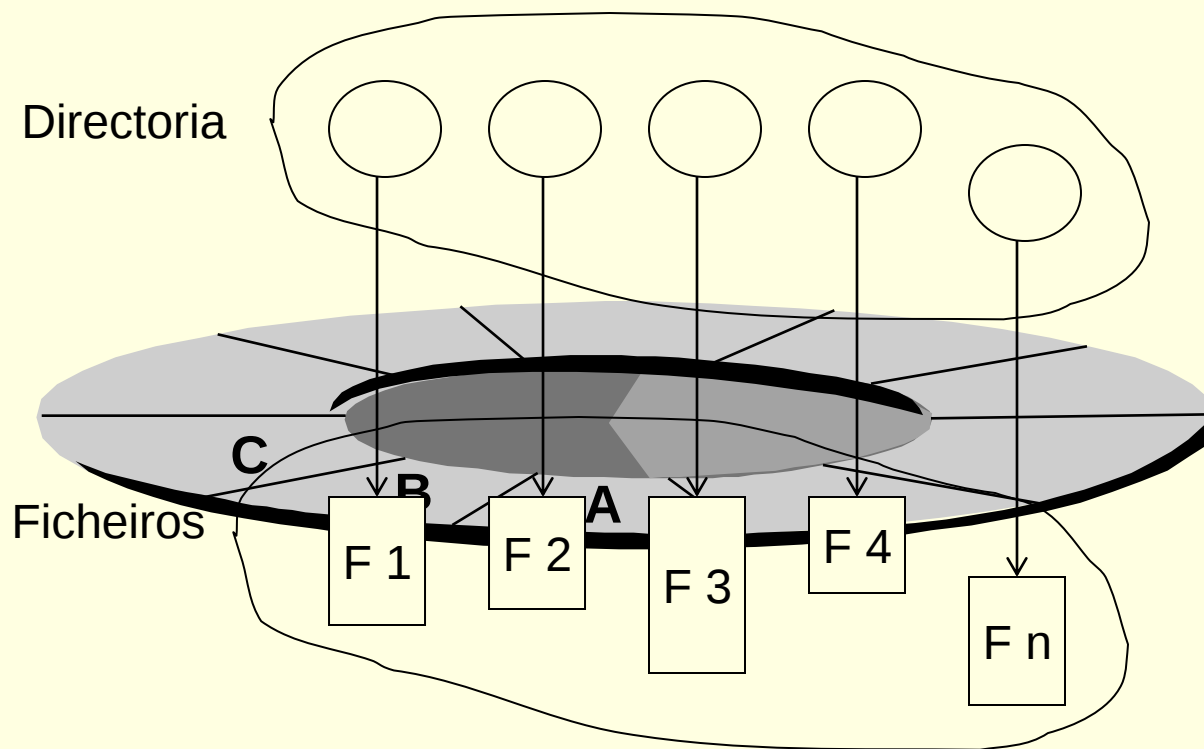
Unix Windows NT Network MacOS DOS/VS Vax/VMS
Linux Solaris HP/UX AIX Mach
Chorus

Parte II: Gestão de Ficheiros

c) Sistema de Ficheiros em Disco –
Organização e Gestão do Espaço (em disco) ...

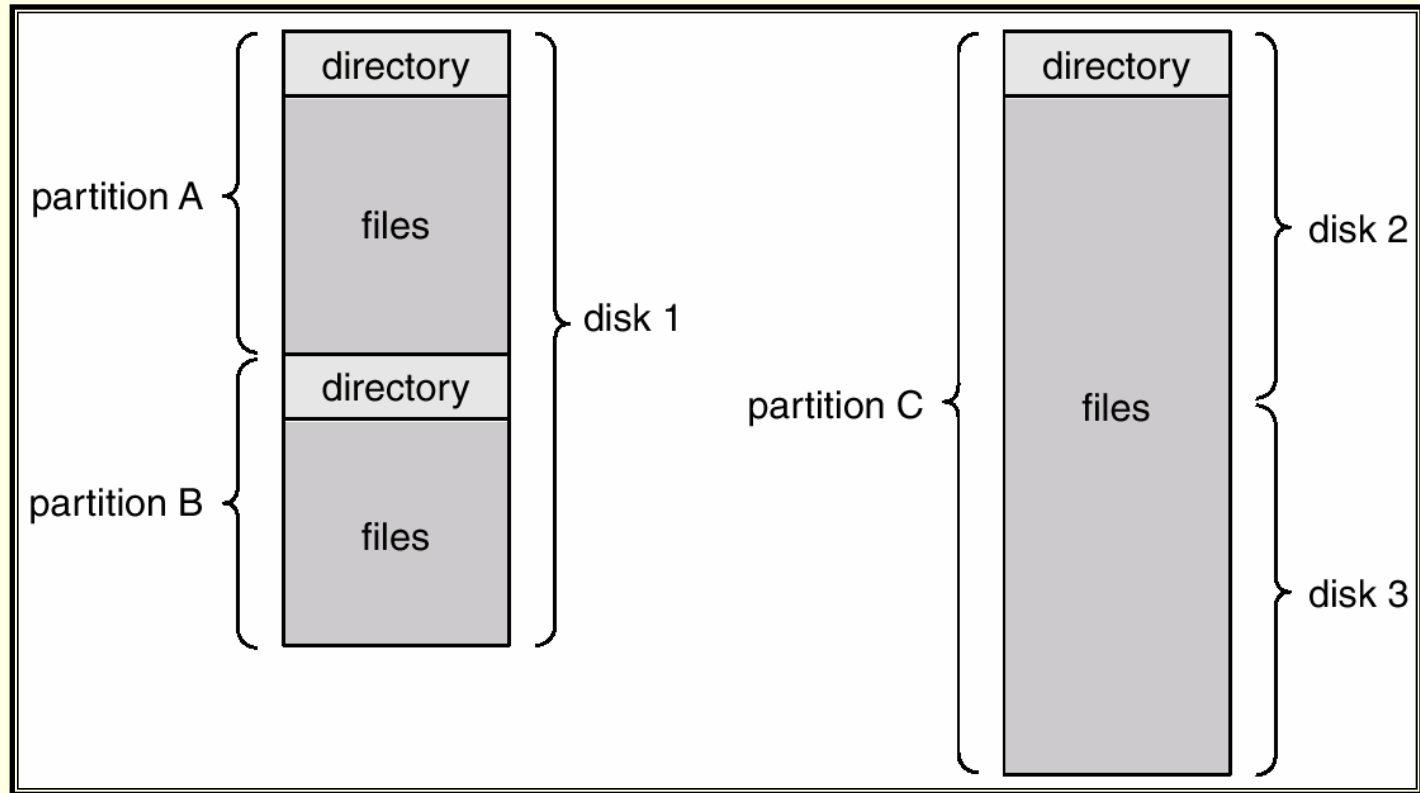
O Sistema de Ficheiros em disco: I

- Como armazenar no disco as “directorias” e os “ficheiros”?



O Sistema de Ficheiros em disco: II

- Um *layout* simples:



O Sistema de Ficheiros “simples”: I

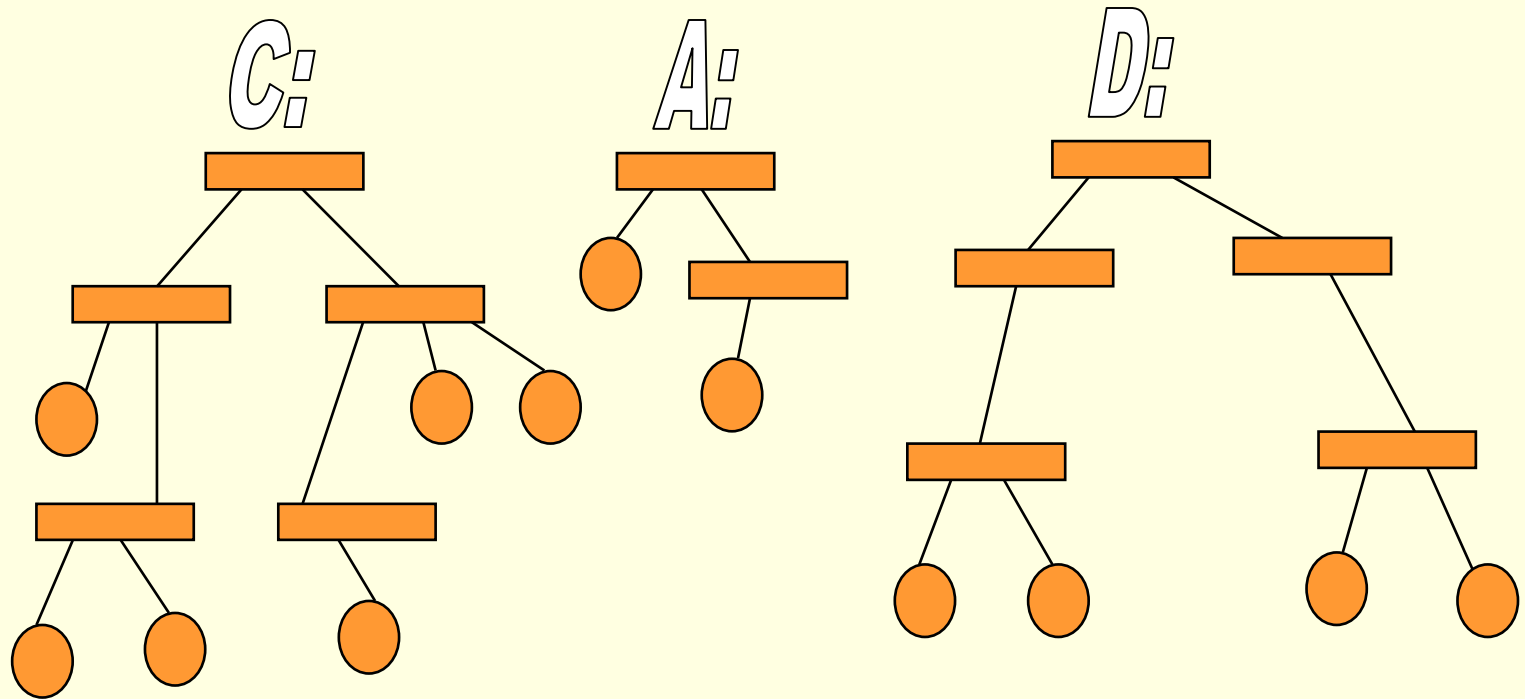
- Para cada partição:
 - Uma estrutura de dimensão fixa (“tabela”) para a directoria
 - Que, por isso, apenas poderá conter um número limitado de entradas com informação sobre os ficheiros existentes nessa partição...
 - Cada entrada pode ser: nome, início, comprimento.
 - Para “informatizar” há que decidir os tipos de dados...
 - `char nome[10]; int inicio, comprimento;`
 - O restante espaço será disponibilizado para os ficheiros...
 - Como geri-lo?

O Sistema de Ficheiros “simples”: Partições

- O que é uma partição?
 - Uma partição é uma parte de um disco físico – formada por um conjunto de blocos contíguos;
 - As partições de um disco são descritas numa estrutura de dados – a Tabela de Partições (geralmente localizada no início do disco)
- Um sistema de ficheiros pode “assentar” as suas estruturas de dados
 - directamente sobre uma partição,
 - caso das partições A e B (fig. da página 3)
 - ou sobre um disco, ou volume lógico (maior detalhe posteriormente)
 - caso da (incorrectamente chamada) partição C

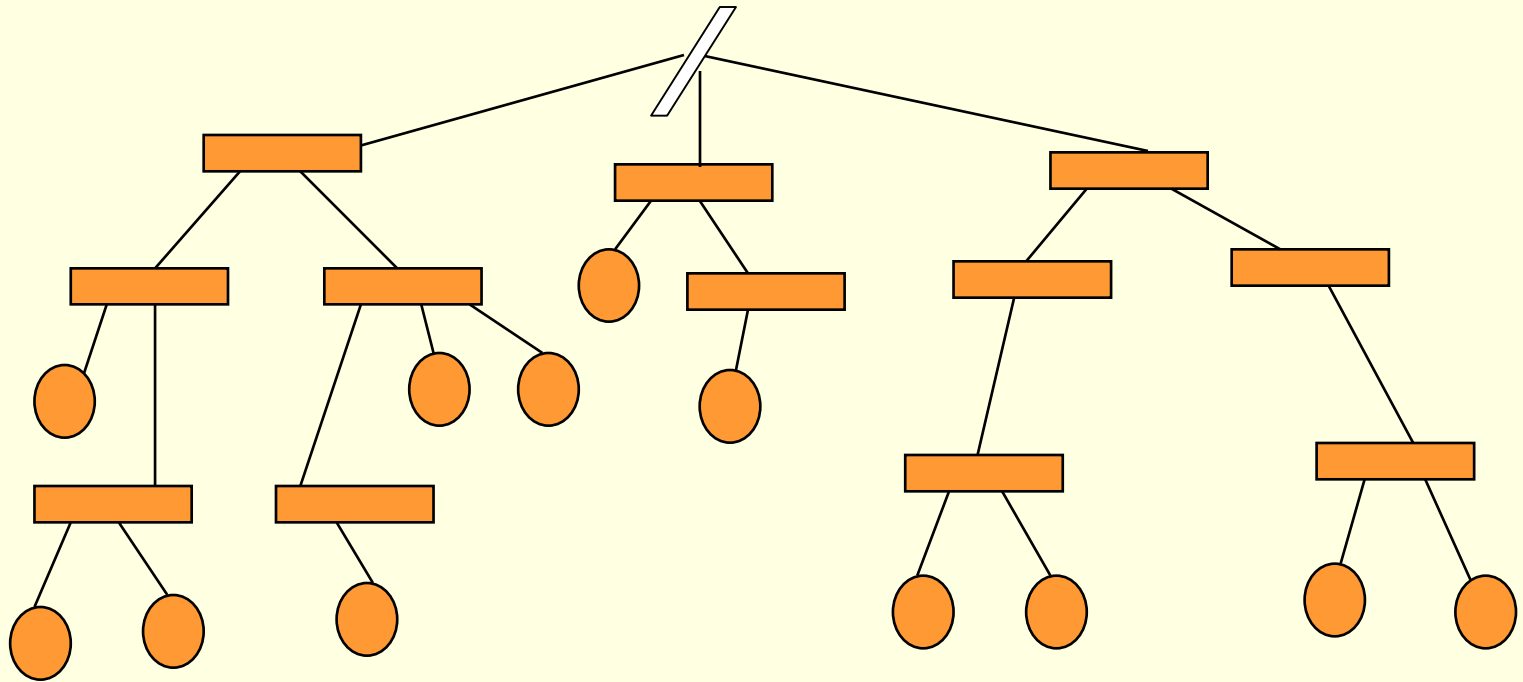
O Sistema de Ficheiros “simples”: II

- Como pode o utilizador ver os múltiplos volumes?
 - Identificados isoladamente (MS-DOS, Win*)



O Sistema de Ficheiros “simples”: III

- Como pode o utilizador ver os múltiplos volumes?
 - “Unidos” numa raiz comum (Unixes, Windows 2000/XP/Vista/...)



Uma directoria “simples”: exercício

- Considere que a directoria é armazenada num único bloco de disco:
 - Uma entrada é...

```
struct entrada {  
    char nome[10];  
    int inicio, comprimento;  
}
```
 - Implemente:
 - Um vector de `struct entrada` capaz de conter o máximo de entradas que podem ser armazenadas num único bloco.

(continua)

Uma directoria “simples”: exercício

(continuação)

- Implemente:
 - Imagine que existe uma função para ler um bloco de disco, dado o seu número:

```
void readBlock(int num, unsigned char * blk)
```
 - Use a função para ler o bloco 1 do disco (bloco que contém a directoria) e carrega a informação no vector de entradas.
 - Faça uma função que percorre esse vector e imprime os nomes e comprimentos dos ficheiros, de uma forma parecida com o comando `dir` do Windows (ou `ls` do Linux).