

# Sistemas Operacionais

Sistema de Arquivos – (parte II de II)

---

# Organização

---

Os sistemas de arquivos dos computadores atuais podem ser “enormes” em volume de dados.

Atualmente, alguns sistemas armazenam centenas de milhares de arquivos em *terabytes* de disco.

Para gerenciar todos estes dados, precisamos organizá-los. Tal organização geralmente é feita em duas camadas distintas:

Partições

Diretórios

# Organização

---

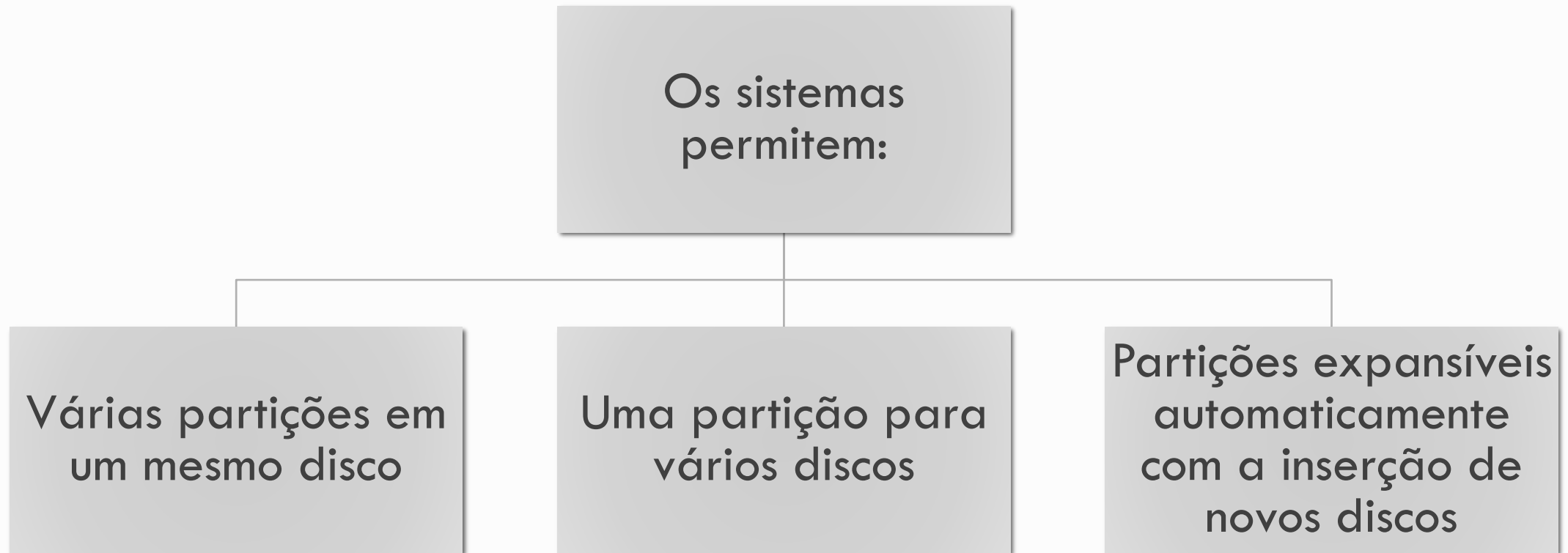


**Partições**, também são chamadas de Minidiscos (IBM) ou Volumes (PCs e Macintosh).

Em geral, cada disco possui pelo menos uma partição, que é uma estrutura de baixo nível onde residem arquivos e diretórios.

# Organização

---



# Organização

---

O usuário então precisa se preocupar apenas com a estrutura lógica de arquivos e diretórios.

Ele pode ignorar completamente os problemas de alocação de espaço físico para os arquivos.

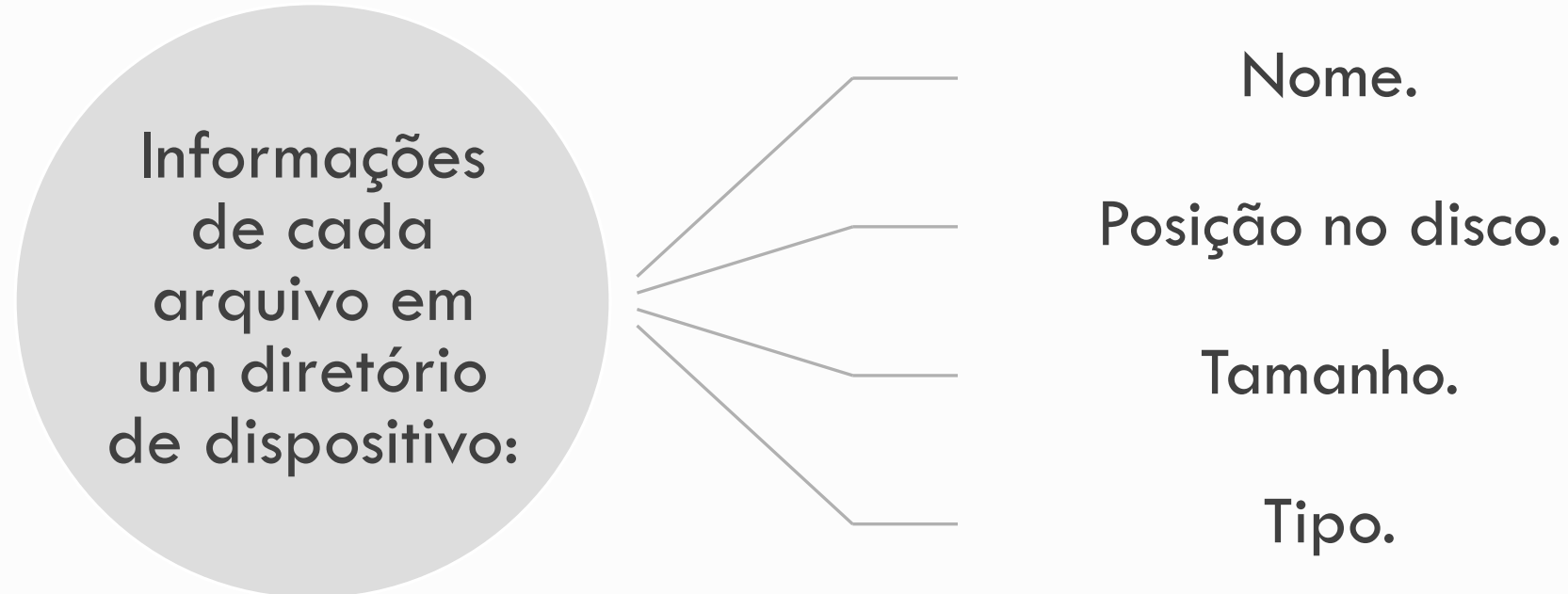
Assim, as partições podem ser consideradas **discos virtuais**.

# Organização

---

Já os diretórios armazenam informações da estrutura dentro de partições específicas.

Tais informações são mantidas em entradas de um diretório de dispositivo ou índice de volume.



# Organização dos Diretórios

Organização em um nível

# Único diretório

---

É a estrutura de diretório mais simples dentro de uma partição é a de apenas um nível.

Todos os arquivos estão contidos no mesmo diretório, que é de fácil suporte e compreensão.

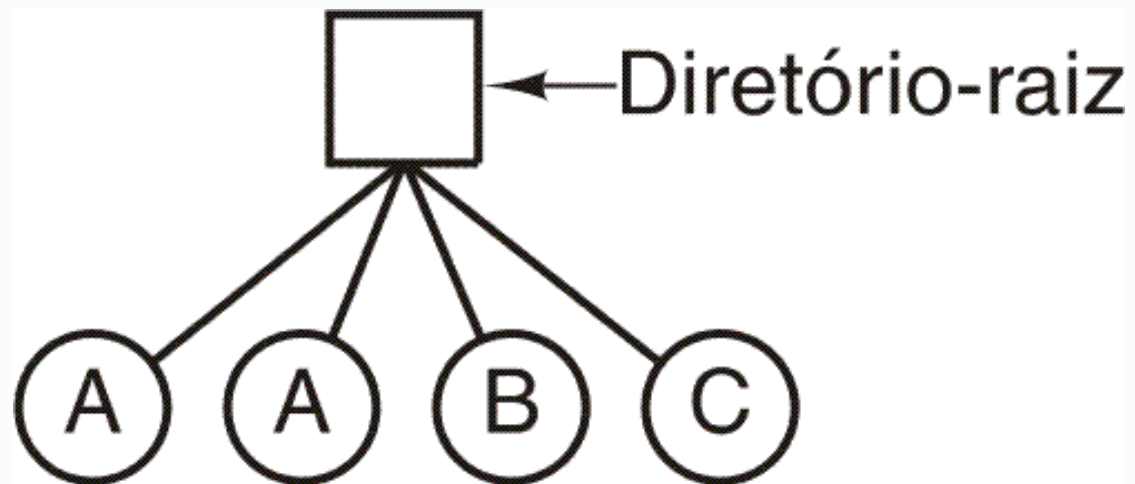
Mas existem limitações...

- Quais?



# Único diretório

---



Exemplo de um sistema de diretório de nível único

- Contém 4 arquivos
- Propriedades de 3 pessoas diferentes, A, B, e C

# Único diretório

---

## **Vantagem:**

Simple implementação.

## **Desvantagens:**

Dificulta a segurança em múltiplos usuários.

Teria que especificar a permissão de cada arquivo.

# Único diretório

---

Organização não é o seu ponto forte.

Imagine que você tenha 5000 arquivos do tipo txt.

Seria complicado nomear arquivos que são similares e estão em um contexto parecido, já que todos precisam ter nomes exclusivos.

Além disso, alguns sistemas operacionais limitavam o tamanho do nome do arquivo (11 caracteres para o MS-DOS e 255 para o UNIX).

# Único diretório

---

Computador que utilizava apenas um diretório:

## 1964 – CDC 6600

- Considerado o primeiro **supercomputador** bem sucedido.
- Ele foi o computador mais veloz do mundo de 1964 até 1969.
- Suporte a multiprogramação.



# Organização dos Diretórios

Organização em dois níveis

# Dois níveis

---

A principal desvantagem de um diretório de nível único é a confusão de nomes de arquivos criadas por diferentes usuários.

## **Imagine:**

Você nunca criou um arquivo com o nome so.docx – mas outro usuário já criou.

Assim, o sistema te retorna um erro indicando que este nome não pode ser mais utilizado.

Como você saberia quais nomes já foram usados por outros usuários?

**A solução foi criar um diretório para cada usuário**

# Dois níveis

---

Nesta estrutura, cada usuário tem seu próprio diretório

- Diretório de arquivo de usuário (UFD – *User File Directory*)

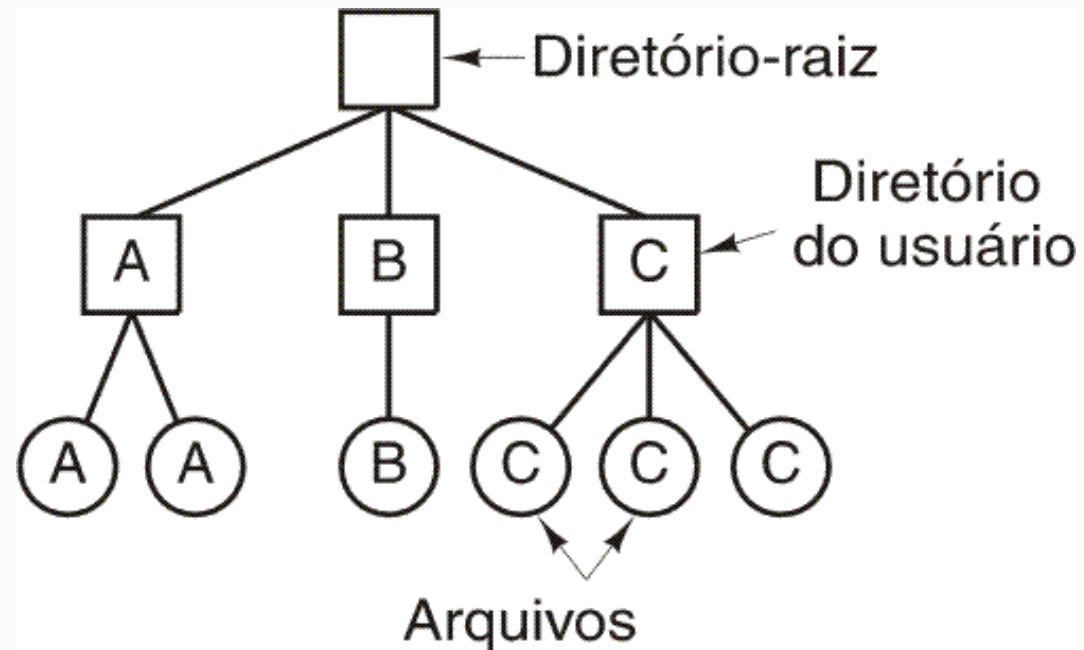
O S.O. controla no processo de login

- O diretório de arquivos mestre (MFD – *Master File Directory*)

Quando um usuário fizer referência a determinado arquivo, apenas seu próprio UFD será pesquisado.

# Dois níveis

---



As letras indicam os donos dos diretórios e arquivos.



# Dois níveis

---

## **Vantagem:**

- Resolve o problema de colisão de nomes.

## **Desvantagem:**

- Não permite que diferentes usuários compartilhem arquivos, pois não existe uma área comum de armazenamento.
- Com o passar dos anos, a quantidade de arquivos de um só usuário tornou-se um problema para esta estrutura.
- Surgiu nesta época a necessidade da organização de arquivos seguindo critérios lógicos.

# Organização dos Diretórios

Organização em árvore

# Organização em árvore

---

Permite a classificação seguindo uma hierarquia lógica a ser definida pelo próprio usuário.

- Fazendo um paralelo:
  - Foi alcançada liberdade semelhante à estrutura dos arquivos (Sequência de bytes), onde o usuário é capaz de criar sua organização dentro do arquivo;

Mais geral que uma árvore, seria um grafo.

- Mas cuidados devem ser tomados em um grafo

# Organização em árvore

---

Todos os modernos sistemas de arquivos são organizados assim em:

Árvore

Grafos

# Organização em árvore

---

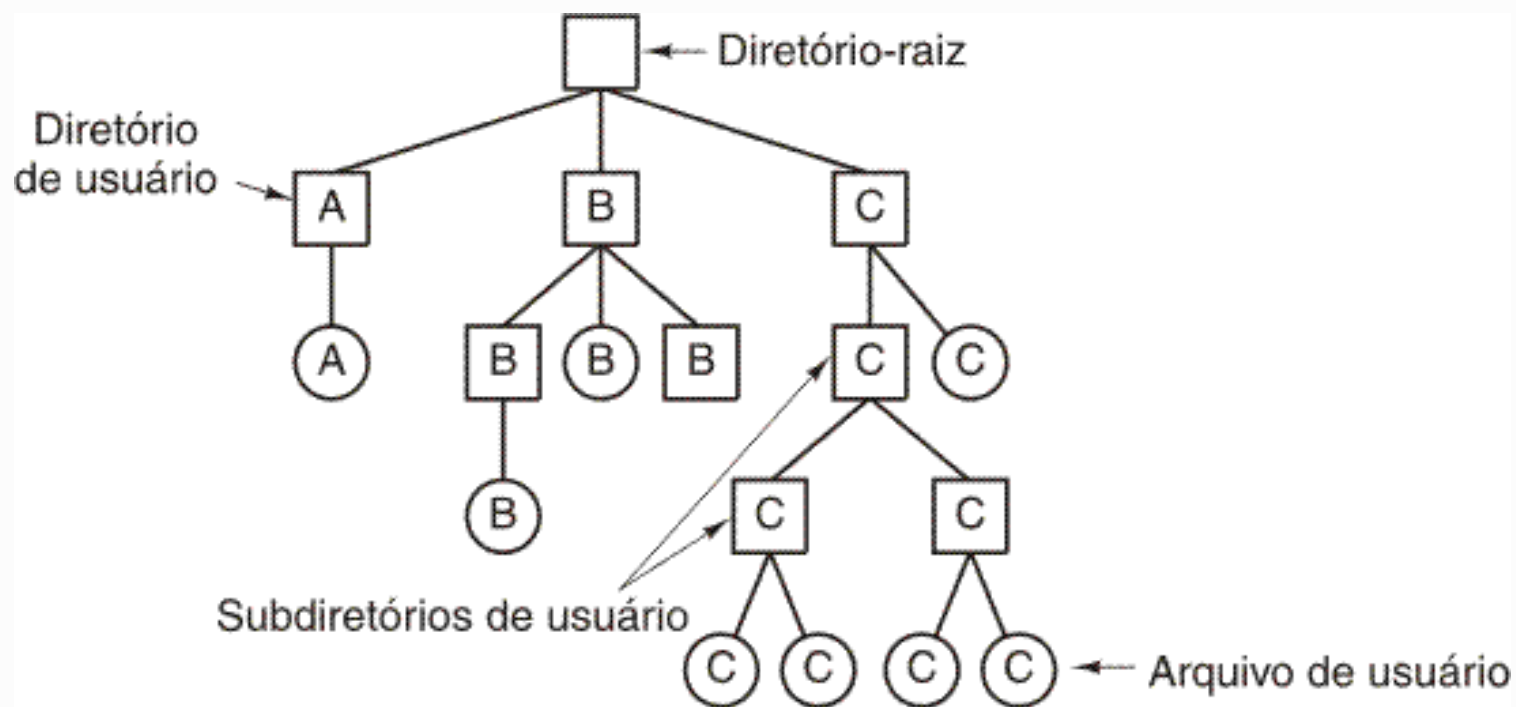
## **Vantagem:**

Classificação hierárquica em qualquer nível desejado.

## **Desvantagem:**

Arquivos podem precisar estar em dois ou mais locais ao mesmo tempo.

# Organização em árvore



Um sistema de diretório hierárquico

# Nomes de Caminhos

Caminho absoluto vs Caminho relativo

# Nomes de caminhos

---

Quando o sistema de arquivos é organizado como uma árvore de diretório, é preciso algum modo de especificar o nome dos arquivos.

São usados comumente dois métodos:

Nome de **caminho absoluto.**

Nome de **caminho relativo.**



# Nomes de caminhos absolutos

---

Os **caminhos absolutos** sempre iniciam no diretório raiz.

Cada S.O. fornece uma forma diferenciada de indicar a separação de diretórios:

- **Windows:** \
- **MAC OS, Unix, Minix e Linux:** /
- **MULTICS:** >

## Exemplo:

- \home\unifal\fotos
- /home/unifal/fotos
- >home>unifal>fotos

# Nomes de caminhos absolutos

---

Se o primeiro caractere do nome do caminho for o separador, então o caminho é absoluto;

- **/home/users/unifal**
- users/unifal
- ./users/unifal



# Nomes de caminhos relativos

---

**Caminho relativo** é usado juntamente com o conceito de diretório de trabalho (Também chamado diretório atual).

Os comandos Unix:

- **cp /home/users/sistemasoperacionais/aula.docx /docs**
- **cp aula.docx /docs**

Possuem o mesmo efeito se o diretório atual for **/home/users/sistemasoperacionais**

# Nomes de caminhos relativos

---

A maioria dos S.O.s que suportam a hierarquia de árvores possuem dois arquivos especiais em cada diretório:

.

..

São arquivos que não podem ser removidos sem a remoção completa do diretório corrente.

- O “ponto” referencia o diretório atual.
- O “ponto-ponto” referencia o diretório pai.

# Nomes de caminhos relativos

---

## Create:

- Cria um diretório “vazio”
- Exceto pelo “.” e pelo “..”
- Programa responsável: **mkdir**



## Delete:

- Remove um diretório vazio
- Vazio quando possui somente “.” e “..”
- Programa: **rmdir**

# Próxima aula

---

Leitura:

Sistemas operacionais modernos

Implementação de Arquivos

# Referências

---

Sistemas Operacionais Modernos. Tanenbaum, A. S. 2ª edição. 2003.

Sistemas Operacionais. Conceitos e Aplicações. A. Silberschatz; P. Galvin; G. Gagne. 2000.

Sistemas Operacionais – Projeto e Implementação. Tanenbaum, A. S. 2ª edição. 2000.

Slides Prof. Humberto Brandão