

**Educação  
Profissional  
Paulista**

Técnico em  
**Ciência de  
Dados**

# **Introdução à Ciência de Dados, ferramentas e jargões da área**

## **Introdução à informática**

### **Aula 2**

**[DADOS]ANO1C1B2S11A2**

# Exposição



## Objetivos da Aula

Introduzir os conceitos de sistemas operacionais.



## Competências da Unidade (Técnicas e Socioemocionais)

- Compreender e dominar os principais conceitos e noções de sistemas operacionais na informática;
- Analisar situações em que cada tipo de sistema operacional pode ser utilizado e identificar os principais equipamentos e finalidades de aplicação para seus usos;
- Trabalhar em grupos.



## Recursos Didáticos

- Recurso audiovisual para a exibição de textos, vídeos e imagens;
- Artigos selecionados para leitura.



## Duração da Aula

50 minutos.

# Conceitos fundamentais de sistemas de arquivo

Os sistemas de arquivos são vitais para os sistemas operacionais, gerenciando o armazenamento, a organização e o acesso aos dados em mídias de armazenamento.

Eles determinam a nomenclatura, a alocação e a segurança dos arquivos dentro de uma estrutura de diretórios. **Conheça cada um:**

NTFS ( <i>New Technology File System</i> )	ext4 ( <i>Fourth Extended File System</i> )	FAT32 ( <i>File Allocation Table 32</i> )
Utilizado pelo Windows, é conhecido por suportar grandes volumes e complexas configurações de permissões, tornando-o adequado para ambientes empresariais.	Popular no Linux, é reconhecido por sua robustez e eficiência no gerenciamento de muitos arquivos pequenos, ideal para servidores e sistemas de banco de dados.	Aloca o primeiro bloco de memória grande, oferecendo um bom equilíbrio entre performance e simplicidade.

Elaborado especialmente para o curso.

# Conceitos fundamentais de sistemas de arquivo

A estrutura de um sistema de arquivos inclui metadados que armazenam informações sobre os arquivos, como:

- tamanho;
- data de criação;
- permissões;
- localização;

Esses dados são **essenciais** para o gerenciamento eficiente dos arquivos e a interação do usuário/aplicativos com o sistema de arquivos.

Além disso, sistemas de arquivos contemporâneos incluem funcionalidades como compressão e criptografia de dados, **otimizando** o uso do espaço e resguardando informações sensíveis.



# Exemplos de conceitos fundamentais de sistemas de arquivo

### Exemplo 1:

O sistema de arquivos NTFS no Windows inclui definir permissões complexas de arquivo e tem suporte para **grandes volumes de armazenamento**, adequado para empresas e servidores.

### Exemplo 2:

O sistema de arquivos **FAT32** é muito usado em dispositivos de armazenamento removíveis, como pen drives, devido à sua compatibilidade universal entre sistemas operacionais, mesmo tendo limitações de tamanho de arquivo e volume.

# Exemplos de conceitos fundamentais de sistemas de arquivo

### Exemplo 3:

No Linux, o **sistema de arquivos ext4** é amplamente utilizado por sua eficiência em gerenciar um vasto número de pequenos arquivos.



### Tome nota

No Linux, o ext4 é ideal para servidores de arquivos e sistemas de bancos de dados.

# Implementação de sistemas de arquivos

A implementação dos sistemas de arquivos define como os dados são escritos e lidos nos dispositivos de armazenamento. Existem métodos de alocação de arquivos, como:

- **Alocação contígua:** simples e eficaz para acessos sequenciais, mas pode causar fragmentação.
- **Alocação encadeada:** utilizada no FAT32, permite fácil expansão dos arquivos, mas acessos aleatórios são mais lentos.
- **Alocação indexada:** usada no ext4, conta com **inodes** que direcionam aos blocos de dados, agilizando o acesso direto aos arquivos.



### Tome nota

**Inodes** são estruturas essenciais em vários sistemas de arquivos UNIX, como o ext3 e ext4 no Linux. Eles guardam dados importantes de um arquivo, menos o nome e o conteúdo. "Inode" vem de "index node".



# Implementação de sistemas de arquivos

Os sistemas de arquivos precisam **gerenciar a integridade e a organização dos dados**, especialmente em situações de falhas do sistema ou desligamentos inesperados.

Sistemas atuais usam o ***journaling***, que anota alterações antes de efetivá-las no sistema principal. Isso permite restaurar o sistema de modo rápido e seguro após falhas, mantendo a consistência dos dados.



### Tome nota

O ***journaling*** é uma técnica em sistemas de arquivos que registra as mudanças em um *log* antes de aplicá-las ao sistema principal. Assim, se houver uma falha durante uma gravação, o sistema operacional verifica o *log* (journal) para finalizar as alterações pendentes e prevenir a corrupção de dados.

# Exemplos de implementação de sistemas de arquivos

### Exemplo 1:

O método de alocação contígua é simples e facilita o acesso sequencial, mas pode levar à fragmentação ao longo do tempo.

Isso pode resultar em ineficiência para sistemas antigos ou sistemas embarcados com necessidades de armazenamento previsíveis.

### Exemplo 2:

Sistemas de arquivos que utilizam alocação encadeada, como o FAT32, permitem a expansão fácil de arquivos adicionando-se mais blocos à cadeia.

Contudo, o acesso aleatório é mais lento, já que cada bloco precisa ser lido em sequência.

# Exemplos de implementação de sistemas de arquivos

## Exemplo 3:

O uso de **inodes** no sistema de arquivos **ext4** permite uma alocação indexada, na qual cada arquivo tem um inode que aponta para seus blocos de dados.



### Tome nota

Alocação indexada eficiente facilita o acesso direto e a expansão de arquivos.

# Gerenciamento de espaço em disco

**Objetivo:** a administração eficaz do espaço em disco é crucial para **maximizar a eficiência do armazenamento** e assegurar a integridade dos dados.

Métodos como *bitmap* ou listas encadeadas são vitais para monitorar os blocos utilizados e disponíveis.

Além disso, sistemas de arquivos com **journaling**, a exemplo do ext4 e NTFS, reforçam a confiabilidade por meio de um log de transações que registra as alterações planejadas nos arquivos.

**Permitindo que:**

👍 O sistema se recupere de forma estável após interrupções.

## Gerenciamento de espaço em disco

O **LVM, ou gerenciamento de volumes lógicos**, oferece uma maneira avançada e flexível de administrar o espaço em disco.

Permite ao administrador do sistema alterar o tamanho de partições e volumes sem interrupções, adaptando o espaço disponível conforme as demandas mudam.

- Vantajoso em servidores, nos quais a necessidade por armazenamento pode mudar bastante.

Os sistemas de arquivos precisam gerir a **fragmentação**: acontece quando arquivos são armazenados em partes dispersas no disco, afetando o desempenho.

Para resolver isso, utilizam-se ferramentas de **desfragmentação** para reorganizar os dados e otimizar o acesso aos arquivos.



### Exemplo

No Windows, desfragmentar é um procedimento de manutenção regular para juntar fragmentos de arquivos e melhorar a velocidade de leitura e escrita no disco.



# Exemplos de gerenciamento de espaço em disco

### Exemplo 1:

Sistemas com ***journaling***, como o ext4, podem recuperar-se rapidamente de falhas, registrando mudanças em um *journal* antes de aplicá-las ao sistema de arquivos principal, garantindo a integridade dos dados após um *crash*.

### Exemplo 2:

O uso do **LVM** em servidores Linux permite o redimensionamento dinâmico de volumes sem interromper o acesso aos dados, facilitando a gestão de espaço em ambientes que requerem flexibilidade de armazenamento.

# Exemplos de gerenciamento de espaço em disco

### Exemplo 3:

A desfragmentação periódica em sistemas de arquivos **NTFS no Windows** reorganiza os dados para melhorar a velocidade de acesso e aumentar o espaço contíguo disponível.



### Tome nota

Essa realização otimiza o desempenho do sistema.

Vamos  
fazer uma  
**atividade**

## Brainstorming: seminário sobre sistemas operacionais mais utilizados no mercado

No seminário sobre sistemas operacionais, exploraremos e debateremos os mais usados atualmente, focando em aprofundar o conhecimento técnico, as aplicações práticas, o perfil dos usuários e o impacto das comunidades de desenvolvedores.

20 min

Em grupo

Cada grupo de alunos será responsável por pesquisar, analisar e apresentar um esboço abrangente que cubra os seguintes pontos:

- 1 Introdução aos sistemas operacionais mais utilizados:** apresentem uma visão geral dos sistemas operacionais dominantes no mercado, incluindo uma breve história e suas estatísticas atuais de uso e popularidade.
- 2 Características técnicas:** analisem as características-chave dos sistemas operacionais dominantes, focando em gestão de memória, processamento, arquivos e segurança e como cada um maneja atualizações e suporte.
- 3 Casos de uso e aplicativos:** identifiquem os perfis de usuários típicos e discutam os casos de uso mais comuns, incluindo aplicações industriais, comerciais e pessoais para cada sistema operacional.
- 4 Desenvolvimento e comunidade:** examinem o modelo de desenvolvimento (código aberto vs. código fechado) de cada sistema operacional, o suporte comunitário e corporativo e como isso influencia sua evolução e adoção.
- 5 Tendências e futuro:** reflitam sobre as inovações recentes e as tendências futuras para os sistemas operacionais, considerando o impacto da computação em nuvem, da inteligência artificial e da Internet das Coisas (IoT).





# Seminário sobre sistemas operacionais mais utilizados no mercado

Durante a apresentação, estejam preparados para responder às seguintes perguntas-chave:

- 1** Quais são as principais diferenças nos recursos de gerenciamento de memória e processamento entre os sistemas operacionais mais utilizados?
- 2** Como cada sistema operacional atende a diferentes tipos de usuários e casos de uso, e quais são as aplicações mais significativas para cada um?
- 3** Qual é o papel das comunidades de desenvolvimento e do modelo de licenciamento na trajetória de sucesso de um sistema operacional?

**Entrega para hoje:** Cada grupo precisa enviar um resumo detalhado da apresentação com fontes usadas no AVA.

**Na próxima aula, cada grupo realizará sua apresentação,** expondo suas descobertas e percepções à turma.



Momento  
de **reflexão**

© Getty Images





O que nós  
**aprendemos**  
**hoje?**

© Getty Images

## Hoje desenvolvemos:

- 1** O entendimento aprofundado sobre **NTFS, ext4 e FAT32**, discernindo como cada um serve a propósitos variados, do NTFS robusto ao FAT32 simples e ao ext4 eficiente no Linux.
- 2** Conhecimentos em gestão de sistemas de arquivos, avaliando métodos de alocação, como **contíguo, encadeado e indexado**, e seu impacto na eficiência de armazenamento e acesso a dados.
- 3** A compreensão do **LVM e do journaling** para garantir a consistência e a recuperação de dados, enfatizando a relevância dessas técnicas para a evolução dos sistemas **operacionais e as demandas por eficácia**.





# Saiba mais

“Qual sistema operacional você usa?”

Você pensa logo no Windows, no terminal do Linux ou no ícone do Android? Mas sistemas operacionais vão além das interfaces gráficas e dos *shells*. Surpreso? Não é só sobre clicar em ícones ou digitar comandos!

Conheça mais no curso de introdução ao sistema operacional.

PESSÔA, C. Sistemas operacionais: conceito e estrutura. *Alura*, 2023. Disponível em: <https://www.alura.com.br/artigos/sistemas-operacionais-conceito-estrutura>. Acesso em: 11 mar. 2024.

# Referências da aula

PESSÔA, C. Sistemas operacionais: conceito e estrutura. *Alura*, 2023. Disponível em: <https://www.alura.com.br/artigos/sistemas-operacionais-conceito-estrutura> Acesso em: 11 mar. 2024.

TANENBAUM, A. S.; BOS, H. *Sistemas operacionais modernos*. São Paulo: Pearson Education do Brasil, 2016.

Identidade visual: Imagens © Getty Images

**Educação  
Profissional  
Paulista**

Técnico em  
**Ciência de  
Dados**