



Arquitetura de Computadores

Gerenciamento de arquivos

Quem sou eu



Júlio César Andrade

Bacharel em Engenharia de Computação - **UEFS**

Especialista em User Experience - **UNIFACS**

Mestrando em Ciências da Computação - **UEFS**

Arquivos

Arquivos podem ser vistos como recipientes que contêm dados ou como um grupo de registros correlatos.



Arquivos

- Os arquivos são implementados através da criação de uma estrutura de dados composta por registros.
- O descritor de arquivo é um registro que mantém informações sobre o arquivo, como posição de início e fim.



Arquivos e diretórios

As informações típicas (atributos) mantidos pelo sistema operacional são:

1. Nome do arquivo
2. Tamanho (bytes)
3. Data e hora da criação, do último acesso, da última modificação
4. Identificação do usuário que criou o arquivo
5. Listas de controle de acesso
6. Local do disco físico onde o conteúdo do arquivo foi colocado

Alocação de arquivos

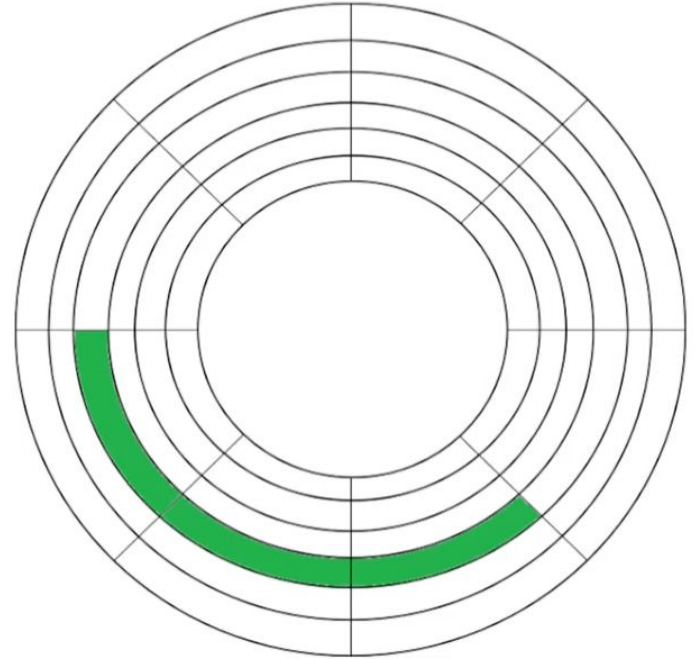
Existem alguns métodos que podem ser utilizados, tais como:

1. Alocação contígua
2. Alocação com Lista Ligada
3. Alocação com Lista Usando um Índice



Alocação Contígua

Este é o esquema mais simples de alocação de arquivos, onde cada arquivo é armazenado no disco como um bloco contíguo de dados.



Alocação Contígua

Por exemplo:

Em um disco com blocos de 1 KB, um pequeno arquivo de 20 KB seria armazenado em 20 blocos consecutivos.



Alocação Contígua

Principais Vantagens:

Simple implementação: Controle de localização de arquivos no disco é feito por um único número (endereço do 1º bloco).

Performance: Leitura do disco é feita de uma vez para todo o bloco (arquivo), requerendo apenas um seek.



Alocação Contígua

Problemas:

Tamanho máximo conhecido: Estratégia requer conhecimento prévio do tamanho máximo do arquivo durante a criação.

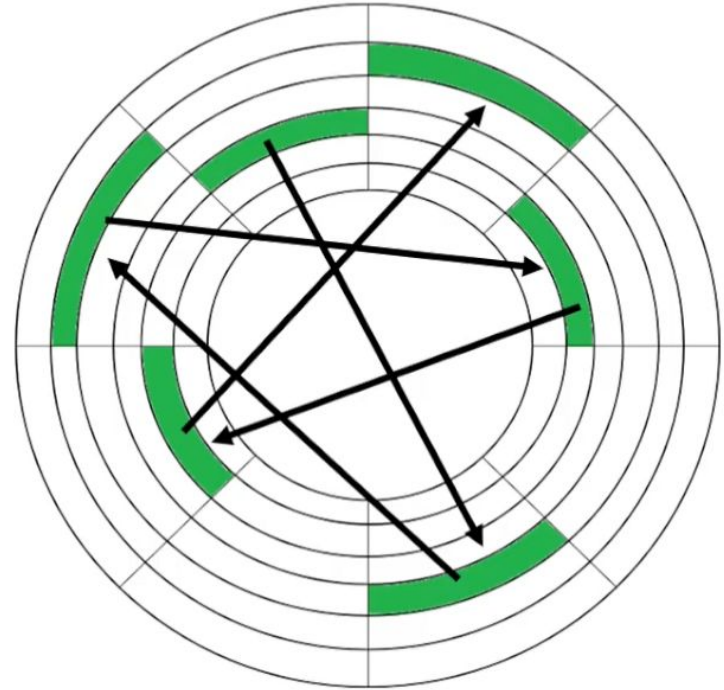
Fragmentação do disco: O esquema causa fragmentação, resultando em lacunas no disco ao remover arquivos, necessitando de compactação (custo alto).



Alocação com Lista Ligada

Nesta estratégia de alocação, usamos uma lista ligada para indicar os espaços ocupados em disco pelo arquivo.

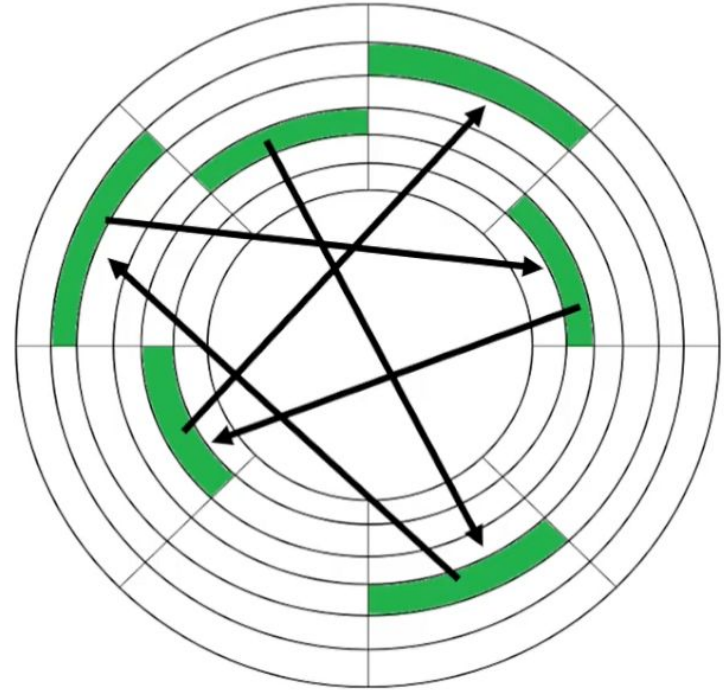
Não é mais necessário que o arquivo seja armazenado em posições contíguas do disco.



Alocação com Lista Ligada

Como funciona?

A primeira palavra de cada bloco é usada com um ponteiro para o próximo bloco e o restante do bloco é usado para armazenar as informações (dados) do arquivo.



Principais Vantagens:

Ausência de fragmentação externa: Não há perda de espaço devido à fragmentação externa.

Flexibilidade de bloco: Qualquer bloco disponível pode ser utilizado, permitindo que os arquivos cresçam indefinidamente enquanto houver espaço no disco.

Estrutura de diretório eficiente: A entrada do diretório só precisa armazenar o endereço do 1º bloco do arquivo, com cada bloco contendo um ponteiro para o próximo bloco do arquivo.



Desvantagens

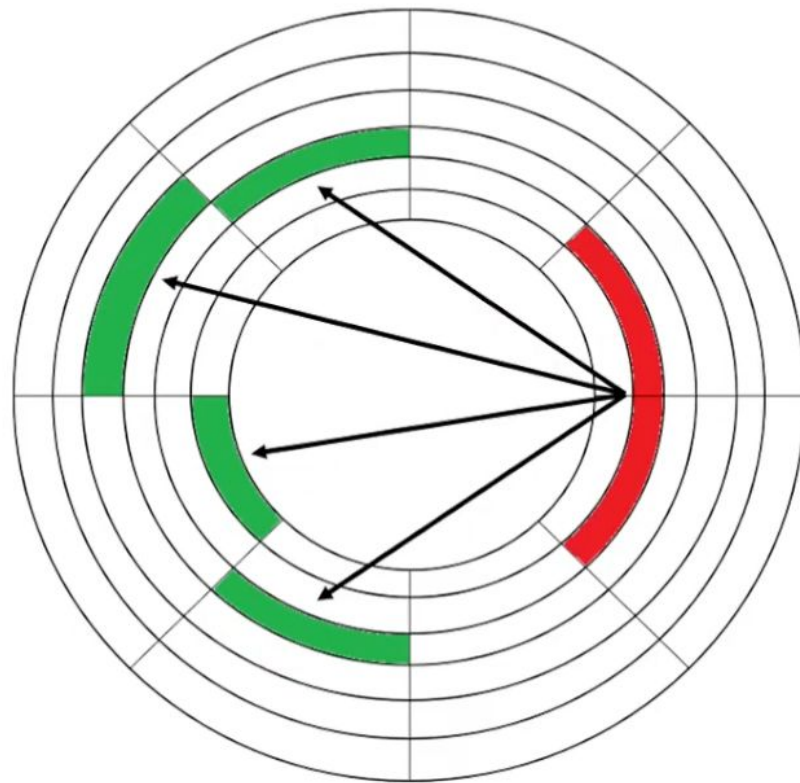
Acesso randômico lento: A necessidade de percorrer a lista torna o acesso randômico lento.

Complexidade na implementação: A implementação deste método de alocação é mais complicada.



Alocação com Lista Usando um Índice

- Resolve o problema de "ponteiros" esparramados pelo disco causado pela alocação encadeada.
- Mantém, por arquivo, um índice de blocos que o compõe.
- O índice é armazenado em um bloco do disco.
- O diretório possui um ponteiro para o bloco que contém o índice associado a um arquivo específico.



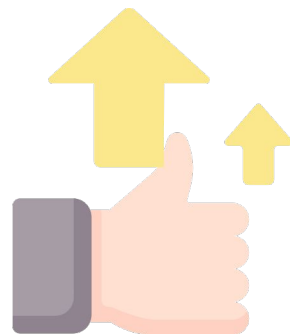
- Tabela
- Arquivos na memória

Vantagens

Elimina desvantagens da alocação com lista ligada.

Remove ponteiros de cada bloco, colocando-os em uma tabela ou índice na memória.

Acesso randômico, mas implementação mais simples em comparação com a lista ligada.



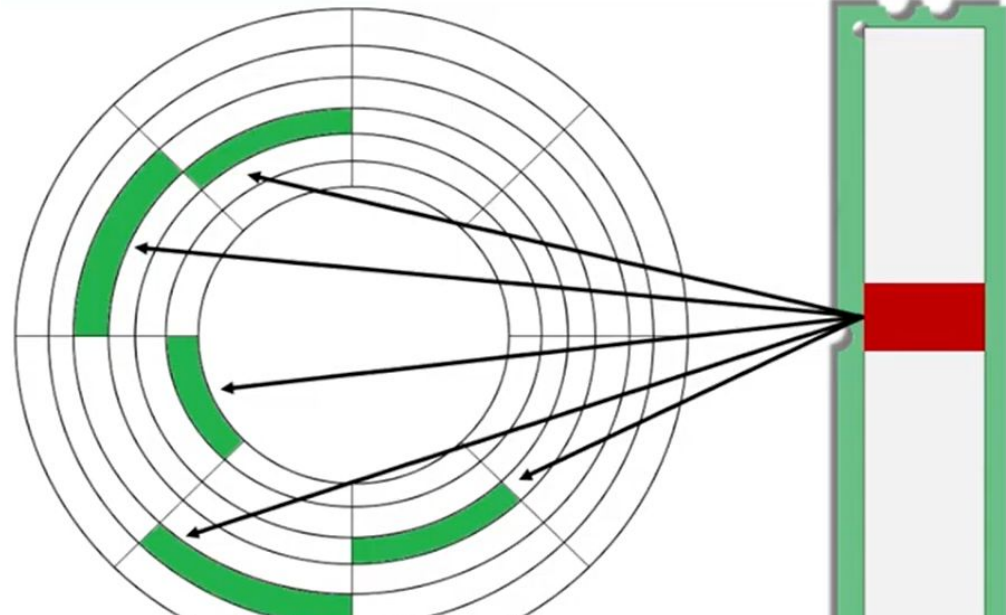
Desvantagens

A tabela também deve estar na memória o tempo todo, o que implica em utilização de espaço de memória.



Alocação por lista usando índice

Retira os ponteiros dos blocos e coloca em um índice na memória principal.



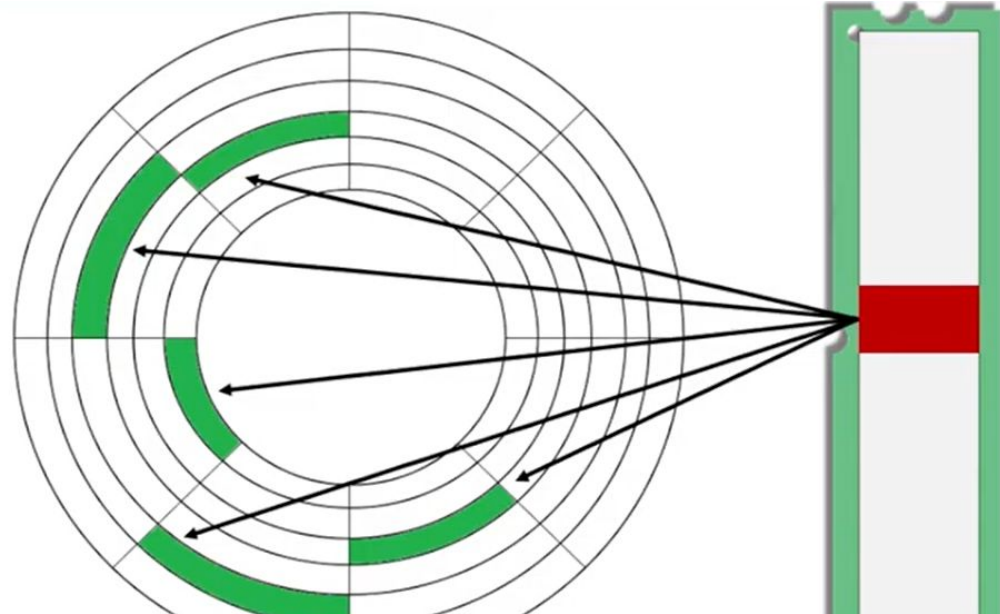
Alocação por lista usando índice

Vantagens:

Ganha mais velocidade em termos de acesso

Desvantagem:

A tabela deve estar sempre na memória principal.



Sistemas de arquivo

Sistema de arquivo (file system) é um conjunto de regras e estruturas usadas para organizar, gerenciar e armazenar informações em HDDs, módulos SSDs, pen drives e cartões de memória.

São essas regras que determinam como os dados serão gravados no dispositivo de armazenamento.

Sistema de arquivo XFS:

- Projetado para sistemas Unix e Linux.
- Padrão em algumas distribuições Linux desde 2014.
- Desenvolvido em 64 bits, compatível com sistemas de 32 bits.
- Reconhecido como um sistema de arquivos de alto desempenho.



Sistema de arquivo HFS:

- Desenvolvido pela Apple para sistemas Mac OS.
- Originalmente usado no Mac OS Standard e no Mac OS Extended.
- Substituído pelo APFS (Apple File System) em sistemas mais recentes.



Sistema de arquivo APFS:

- Desenvolvido pela Apple para substituir o HFS+.
- Otimizado para eficiência de espaço e suporte a metadados avançados.
- Inclui criptografia integrada e permite a criação de snapshots.
- Projetado para melhor desempenho em dispositivos de armazenamento flash.
- Compatível com múltiplas plataformas da Apple, como macOS, iOS, watchOS e tvOS.



Sistema de arquivo EXT3

- Uma das extensões do sistema de arquivos EXT para Linux.
- Sucessor do EXT2 e predecessor do EXT4.
- Suporta journaling para maior segurança e recuperação de falhas.
- Exemplo: Amplamente usado em distribuições Linux mais antigas.



Sistema de arquivo FAT32

Sistema de arquivo FAT de 32 bits.

Suporta arquivos de até 4 GB.

Amplamente utilizado em dispositivos removíveis.

Exemplo: Comum em pen drives e cartões de memória.



Sistema de arquivo NTFS:

- Padrão no sistema operacional Microsoft Windows.
- Aceita volumes de até 2 TB.
- Suporta tamanho de arquivo limitado apenas pelo volume.
- Reconhecido por sua segurança e facilita a recuperação de erros com maior facilidade.



Journaling

Técnica utilizada para melhorar a integridade do sistema de arquivos em caso de falhas ou interrupções inesperadas.

Processo de Journaling:

- Mantém um registro (journal) das operações a serem realizadas antes de efetivamente aplicá-las ao sistema de arquivos.
- As alterações no sistema de arquivos são agrupadas em transações.
- Uma transação é considerada atômica, o que significa que é tratada como uma operação única e completa.

Processo de Journaling:

- Quando uma operação é solicitada (por exemplo, gravação de arquivo), a operação é registrada no journal antes de ser executada no sistema de arquivos.
- Em caso de falha (como queda de energia), o sistema pode reconstruir o estado consistente do sistema de arquivos usando o journal.

Processo de Journaling

Vantagens:

Minimiza a possibilidade de corrupção de dados.

Facilita a recuperação do sistema para um estado coerente após falhas.

Desvantagem:

Pode haver um leve overhead de desempenho devido à gravação adicional no journal, mas é considerado um compromisso aceitável para melhorar a robustez do sistema.

Bibliografia

SILBERSCHATZ, A. & GAGNE, G. & GALVIN, P. B. **Fundamentos de Sistemas Operacionais**. Rio de Janeiro, 2004.

TANENBAUM, A.S. **Sistemas Operacionais Modernos**. 2ª.ed. São Paulo, 2009.

