

Conceitos de Sistemas de Arquivos e implementação didática simulando alocação de espaço em disco

Lucas Roman Lise, Elias Sartori, Renan Sczcepanik

Curso de Bacharelado em Ciências da Computação – Universidade Regional
Integrada do Alto Uruguai e das Missões (URI) – Campus de Erechim
99709-910 – Erechim – RS– Brasil

`elias.sartori@gmail.com, lucas_lisel@hotmail.com, renan.skky@gmail.com`

Resumo. *O presente artigo tem por objetivo apresentar alguns conceitos básicos de arquivos e diretórios. Porém dando mais ênfase nas principais características de alocação em disco de um sistema de arquivos, podendo ser de diferentes segmentações como: alocação contígua, alocação indexada e alocação por lista encadeada, visando expor suas principais características, vantagens e desvantagens. É abordado também a implementação didática realizada com o intuito de visualizar o processo mais facilmente.*

1.0 Introdução

Sistemas de Arquivos é a parte de um Sistema Operacional responsável por gerenciar os arquivos, proporcionando ao usuário uma leitura dos dados e organizando as informações.(TANENBAUM; WOODHULL, 2000). É a parte mais visível do Sistema Operacional, pois está em constante envolvimento com o usuário. Independente dos diferentes tipos de dispositivos de armazenamento deve agir da mesma maneira.

Em um Sistema Operacional a criação de arquivos em um dispositivo de armazenamento exige uma constante organização das informações

presentes nos dados. Essa organização é feita geralmente através de listas ou tabelas, possibilitando a identificação dos blocos livres.

Visando o entendimento do usuário, abordaremos uma ferramenta visual com o intuito de apresentar como é feito o monitoramento e os principais tipos de Alocações do Espaço em Disco.

2.0 Arquivos

Identificados por uma sequência de caracteres, é um método de abstração, oferecendo ao usuário a possibilidade de acesso às informações no disco, ocultando os detalhes de como realmente ocorre o armazenamento.(TANENBAUM; WOODHULL, 2000).

Em alguns sistemas a identificação do arquivo é dividida em duas partes, o nome, e a extensão que servirá para identificar seu conteúdo.

3.0 Informações Básicas

As informações dentro de um computador digital são representadas através de números, os arquivos recebidos são tratados como Streams de Bytes, o Byte é um número entre 0 e 255 que representa os símbolos do computador.(TANENBAUM; WOODHULL, 2000).

I	N	F	O	R	M	A	T	I	C	A
73	78	70	79	82	77	65	84	73	67	65

Figura 1 - Exemplo de uma stream de bytes da palavra informática

4.0 Diretórios

A organização dos arquivos em um sistema de arquivos é feita através de diretórios e eles também são arquivos.(TANENBAUM; WOODHULL, 2000).

- **Nível Único**

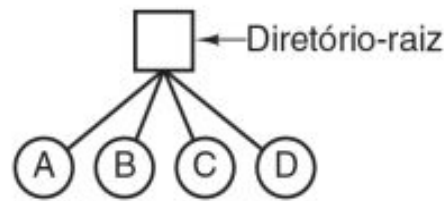


Figura 2 - Exemplo de Diretório de Nível Único

1. Implementação mais simples
2. Contém apenas um diretório organizando todos arquivos do disco
3. Não é possível criar arquivos com o mesmo nome

- **Múltiplos níveis ou árvores**

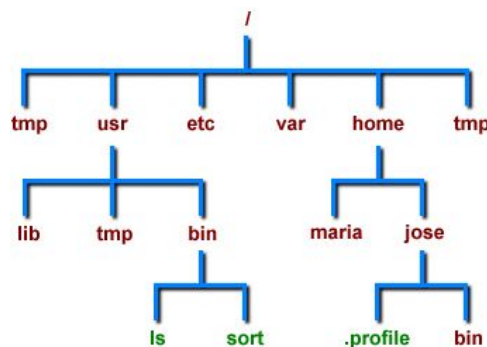


Figura 3 - Exemplo de Diretório com Múltiplos Níveis

1. Usado pela maioria dos Sistemas Operacionais
2. Cada usuário pode criar quantos diretórios forem necessários
3. Em alguns sistemas o diretório só pode ser excluído se estiver vazio

4. Alocação do Espaço em Disco

A criação de arquivos exige que o sistema operacional tenha organizado quais blocos estão livres no disco, essa organização é realizada através de

estruturas geralmente listas ou tabelas, possibilitando ao sistema de arquivos identificar os espaços livres e conhecer quais blocos no disco acompanham quais arquivos.(TANENBAUM; WOODHULL, 2000).

Existem vários métodos de implementação de sistemas de arquivos para diferentes sistemas operacionais, a seguir veremos alguns desses métodos e sua respectiva ferramenta para auxílio.

4.1 Alocação Contígua

A alocação contígua consiste em armazenar um conjunto de blocos sequencialmente dispostos, não há um bloco vazio entre blocos do mesmo arquivo e permite ao sistema localizar um arquivo através do endereço do primeiro bloco e da sua extensão em blocos.(TANENBAUM; WOODHULL, 2000).

É simples de implementar, fácil para monitorar os blocos de um arquivo, pois é somente identificar o endereço de disco do primeiro bloco e seu tamanho e tem um excelente desempenho, pois um arquivo inteiro pode ser lido em apenas uma operação.

Possui a desvantagem do tamanho do arquivo ter de ser conhecido no momento da gravação ou seria viável se o arquivo fosse gravado em uma só vez. Fragmentação do disco. Necessidade de desfragmentação periódica.

4.1.1 Estratégias de Alocação

Existem três técnicas utilizadas para alocar um arquivo contíguo, o First-fit consiste em usar o primeiro espaço livre que ele encontrar no disco, o Best-fit que se concentra em buscar o menor espaço livre para alocar o arquivo e o Worst-fit que usa o maior espaço livre disponível para alocar o arquivo.(TANENBAUM; WOODHULL, 2000).

4.1.2 Fragmentação

Na Figura 4 podemos observar a evolução da fragmentação em um disco, onde arquivos são adicionados e removidos, gerando espaços em branco, dificultando a alocação de novos arquivos.

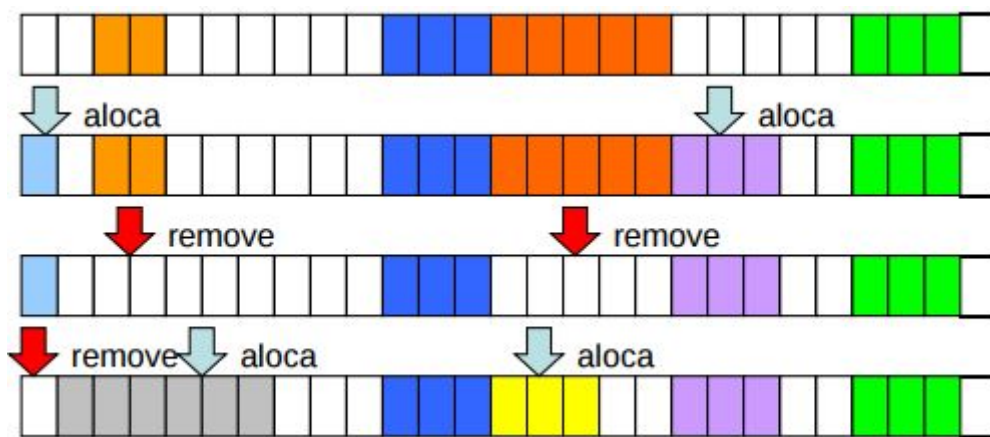


Figura 4 - Exemplo da evolução da fragmentação



Figura 5 - Exemplo de um disco fragmentado

4.1.3 Desfragmentação

Tem como objetivo solucionar o problema da fragmentação e deve ser realizada periodicamente, consiste em agrupar os fragmentos no disco, gerando espaços para arquivos com maior tamanho.(TANENBAUM; WOODHULL, 2000) Alguns algoritmos são usados para que a

desfragmentação seja mais eficiente, a Figura 5 retrata algumas estratégias usadas.

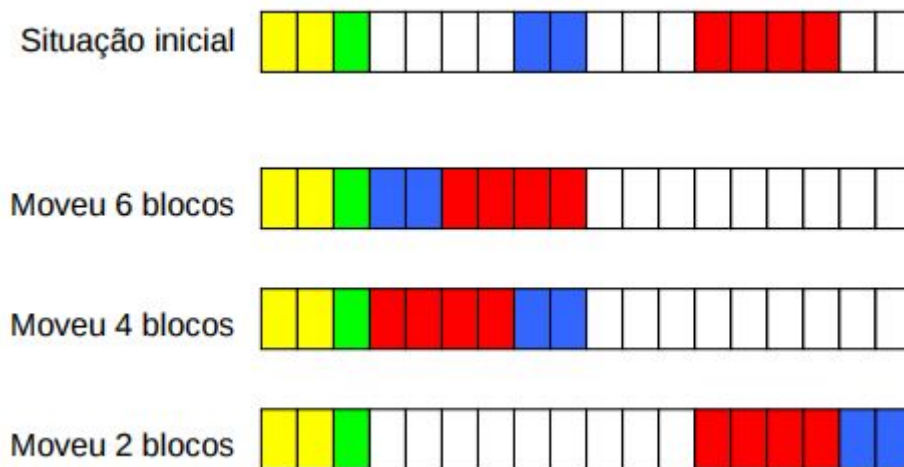


Figura 5 - Exemplo de estratégias de desfragmentação

4.2 Alocação por Lista Encadeada

Tem como características a primeira palavra de cada bloco ser utilizada como ponteiro para a seguinte, todos os blocos do disco podem ser utilizados, não haverá desperdício de espaço com a fragmentação de disco e por isso, ocorre a fragmentação de arquivos, devido a possibilidade de os blocos não serem contíguos, essa fragmentação de arquivos é chamada de extends.(TANENBAUM; WOODHULL, 2000).

Possui a vantagem de todo o disco ser utilizado, não tem fragmentação externa, o tamanho dos arquivos podem ser flexíveis e só necessita do primeiro bloco para encontrar os outros, pois eles estão interligados por ponteiros.

Este sistema fica muito lento quando é requerido acesso aleatório, ele desperdiça espaço nos blocos com o armazenamento de ponteiros e ele só permite acesso sequencial ao blocos.

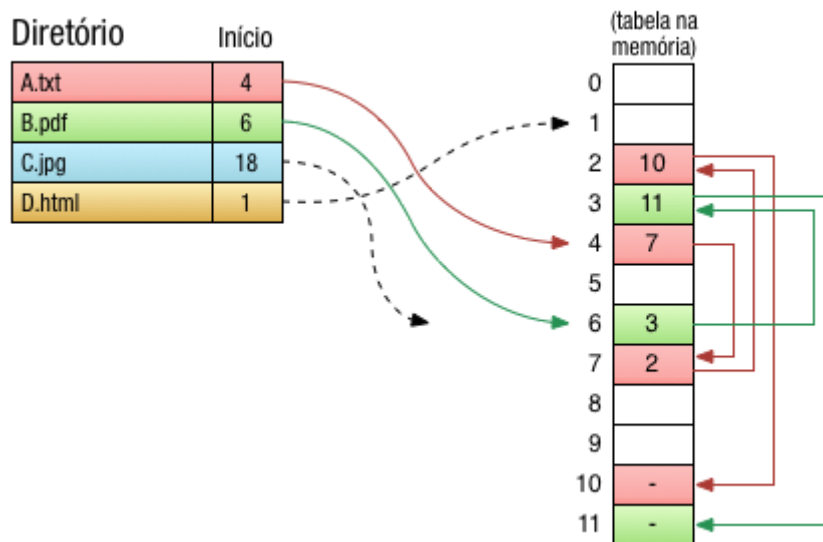


Figura 6 - Exemplo de alocação por lista encadeada

4.3 Alocação Indexada

A alocação indexada usa uma tabela de ponteiros que indicam o bloco no qual os arquivos se encontram e a cada novo bloco ocupado, se adiciona um novo registro na tabela.(TANENBAUM; WOODHULL, 2000).

Esse tipo de alocação possui a vantagem de ter um bloco inteiro disponível para os dados, o acesso aleatório é facilitado, tem a cadeia de localização presente na memória, não referenciando o disco e é suficiente armazenar o endereço do bloco de índice para encontrar o restante dos blocos, não importando o tamanho do arquivo.

O único problema notável que esse tipo de alocação apresenta é um gasto excessivo de memória principal.

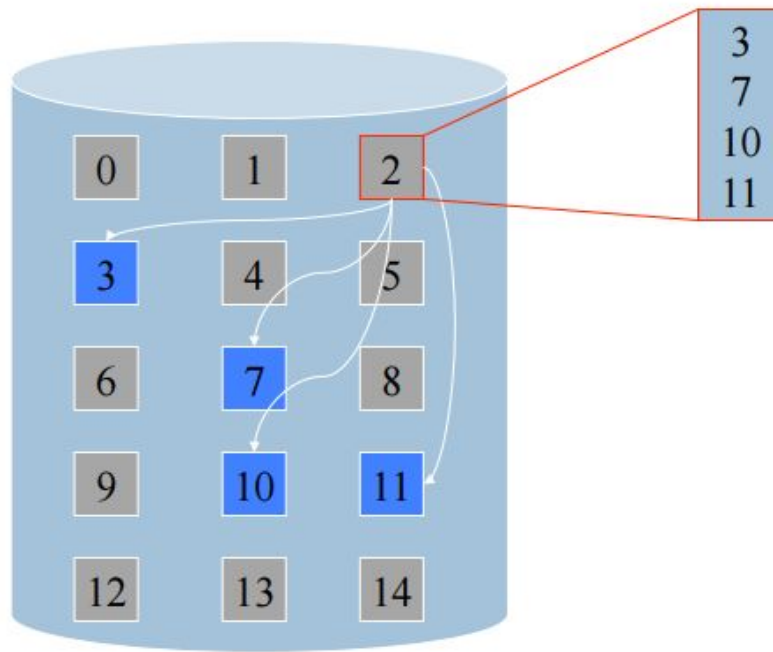


Figura 7 - Exemplo de Alocação utilizando um bloco de índice na memória

5.0 Implementação didática

Foi desenvolvida uma implementação didática na linguagem *Ruby*, utilizando o framework *Ruby on Rails*. Ela foi projetada com o objetivo de demonstrar de forma fácil e visual para um usuário leigo como funciona a alocação de dados em disco a partir de um sistema de arquivos. Segue abaixo as particularidades de cada alocação implementada, sendo estas alocação contígua(Figura 8), alocação encadeada(Figura 9), alocação indexada(Figura 10).



Figura 8 - Alocação Contígua com disco fragmentado.



Figura 9 - Alocação Encadeada.

Ajuda

Voltar

Alocação Indexada

* Tamanho bloco

1

* Tipo bloco

D

Gravar

* Deletar

A

Deletar

Disco

Tabela

#	Blocos
0	1,2,3,4,5,6,7
1	A
2	A
3	A
4	A
5	A
6	A
7	A
8	9,10,13,14,15
9	B
10	B
11	12
12	C
13	B
14	B
15	B
16	
17	
18	
19	

Diretório

Tipo Bloco	Bloco de Índice
A	0
C	11
B	8

Disponível: 4 blocos

Usado: 16 blocos

Restaurar

Figura 10 - Alocação Indexada.

Considerações finais

Como foi possível abordar no artigo, cada tipo de alocação possui a sua particularidade. Falando primeiramente da alocação contígua, suas principais características são: é simples de implementar, e a leitura é realizada com apenas uma operação, porém necessita de desfragmentações periódicas pois o tamanho de cada arquivo é fixo. Quando se trata de alocação encadeada, suas vantagens são: Não existe fragmentação, todo o disco pode ser usado para armazenamento. Suas principais desvantagens: o acesso ao arquivo pode se tornar lento, isto ocorre pois seus blocos estão dispersos na memória, também é desperdiçado espaço para o armazenamento dos ponteiros. Levantando os os pontos da alocação indexada suas vantagens são: um bloco inteiro é reservado para salvar as posições dos blocos, assim facilitando o acesso aleatório.

Com isso é possível concluir que a alocação contígua é mais simples de implementar porém necessita ser desfragmentada periodicamente, enquanto a alocação encadeada e indexada não são fragmentadas e usam todo o dispositivo de armazenamento, porém desperdiçam mais espaço.

Referências:

TANENBAUM, Andrew S.; WOODHULL, Albert S.. **Sistemas operacionais: Projeto e implementação**. 2 ed. Porto Alegre: Bookman, 2000. 751 p.