

Conceitos de Sistemas de Arquivos com ênfase na alocação de arquivo em disco

Lucas Roman Lise - lucas_lise1@hotmail.com

Elias Sartori - elias.sartori@gmail.com

Renan Sczcepanik - renan.skky@gmail.com

Professor: Marcos Lucas

Disciplina: Sistemas Operacionais - Ciência da Computação
Departamento de Ciências Exatas e Engenharias - Universidade Regional Integrada
do Alto Uruguai e das Missões (URI).
Av. Sete de Setembro, 1621, Fátima CEP 99709-510 - Erechim - RS - Brasil

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 contínua, alocação indexada e alocação por lista encadeada. Também é abordado o conceito de desfragmentação.

Palavras-chave: Sistema de arquivos, alocação, desfragmentação, lista encadeada, indexação.

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. 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.

1. 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.

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

2. Informações Básicas

- As informações dentro de um computador digital são representadas através de números
- Arquivos são concebidos e tratados como Streams de Bytes
- Byte é um número entre 0 e 255, exemplo Figura 1.

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

3. Diretórios

A organização dos arquivos em um sistema de arquivos é feita através de diretórios, que em muitos em muitos sistemas também são arquivos.

- **Nível Único**

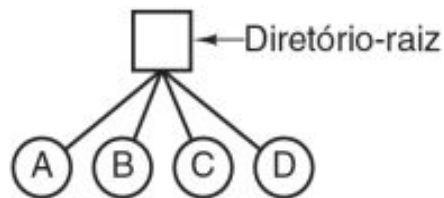


Figura 2 - Exemplo de Diretório de Nível Único

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**

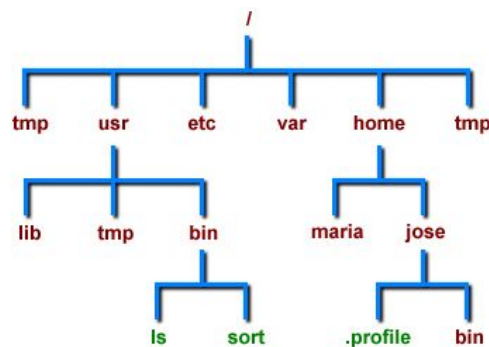


Figura 3 - Exemplo de Diretório com Múltiplos Níveis

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.

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

Características:

- Consiste em armazenar um conjunto de blocos sequencialmente dispostos
- Não há um bloco vazio entre blocos do mesmo arquivo
- Permite ao sistema localizar um arquivo através do endereço do primeiro bloco e da sua extensão em blocos

Vantagens:

- 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

- Excelente desempenho, pois um arquivo inteiro pode ser lido em apenas uma operação

Desvantagens:

- Tamanho do arquivo deve 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

- **First-fit:** usar o primeiro espaço livre
- **Best-fit:** usar o menor espaço livre
- **Worst-fit:** usar o maior espaço livre

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.

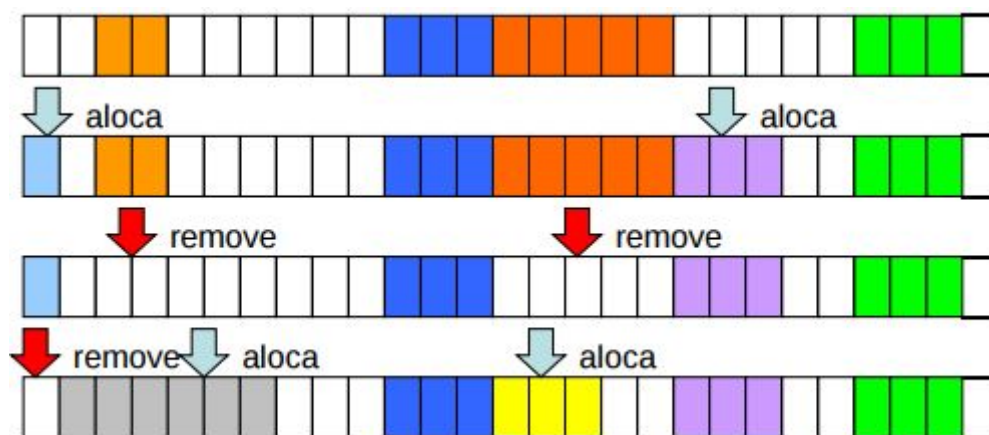


Figura 4 - Exemplo da evolução da fragmentação



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. Alguns algoritmos são usados para que a desfragmentação seja mais eficiente, a Figura 5 retrata algumas estratégias usadas.

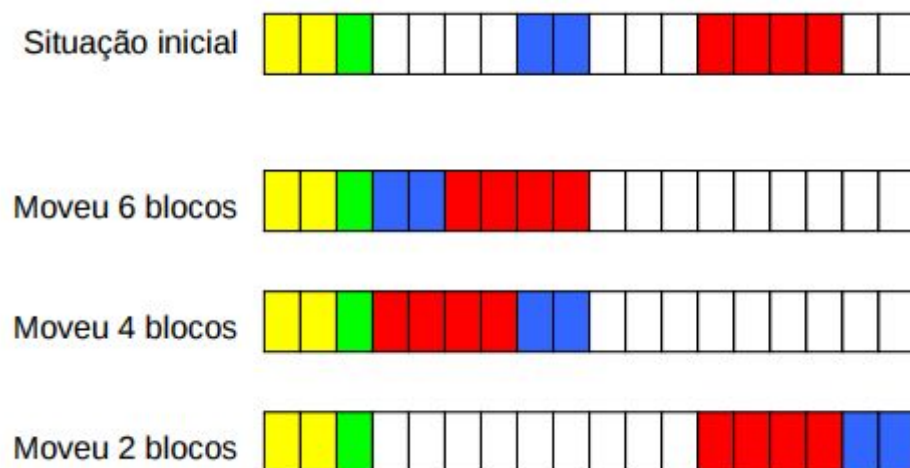


Figura 5 - Exemplo de estratégias de desfragmentação

4.2 Alocação por Lista Encadeada

Características

- A primeira palavra de cada bloco é utilizada com ponteiro para a seguinte
- Todos os blocos do disco podem ser utilizados
- Espaços não serão desperdiçados em fragmentação de disco
- Ocorre a fragmentação de arquivos, devido aos blocos não precisarem ser contíguos, podem ser quebrados e são denominados extends

Vantagens

- Todo o disco pode ser usado
- Não há fragmentação externa
- Tamanho dos arquivos podem ser flexíveis, podendo variar
- É suficiente armazenar o endereço do primeiro bloco para encontrar o restante, pois os ponteiros vão auxiliar

Desvantagens

- Acesso aleatório é extremamente lento
- Desperdiça espaço nos blocos com o armazenamento de ponteiros
- Somente permite somente acesso sequencial aos blocos

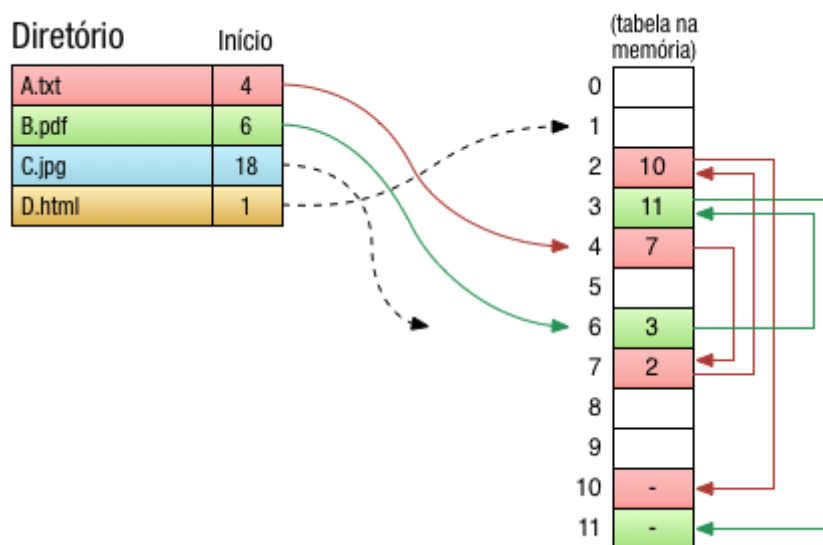


Figura 6 - Exemplo de alocação por lista encadeada

4.3 Alocação Indexada

Características

- Cada palavra de ponteiro de cada bloco de disco é colocado em uma tabela ou em um bloco de índice na memória
- A cada novo bloco ocupado, inclui-se um novo item na tabela ou bloco, exemplo na Figura 7
- Ao acessar o arquivo o bloco de índice é carregado na memória principal

Vantagens

- O bloco inteiro estará disponível para dados
- Acesso aleatório é mais fácil
- Não faz qualquer referência no disco, pois a cadeia para localização está toda presente na memória
- É suficiente armazenar o endereço do bloco de índice para encontrar o restante, independente do tamanho do arquivo

Desvantagens

- Gasto excessivo de memória principal

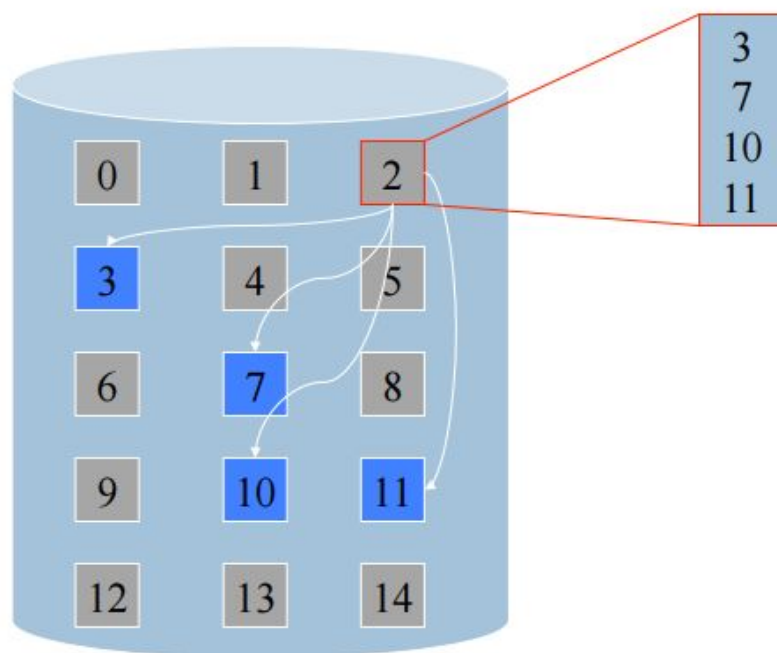


Figura 7 - Exemplo de Alocação utilizando um bloco de índice na memória

5.0 Gerenciamento de espaço em disco

No momento em que os arquivos são armazenados em disco é necessário analisar alguns fatores, dentre eles como os bytes do arquivo serão alocados, isto de forma consecutiva ou divididos e Links de pesquisa: m blocos consecutivos. Deve

considerar que armazenando os arquivos de forma contínua arquivos que no decorrer do tempo atingir um grande porte necessitarão ser alocados novamente no disco, levando em consideração este fato, a maioria dos sistemas de arquivos dividem os arquivos e utilizam blocos de tamanhos fixo.

5.1 Tamanho de bloco

Quando é introduzido o conceito de blocos de arquivos é necessário definir qual será o tamanho deste bloco. Utilizando blocos fixos com um tamanho razoavelmente grande no disco gera um possível problema de desperdício de espaço, pois um arquivo minimamente pequeno obrigatoriamente terá que utilizar um bloco inteiro. Em contrapartida blocos demasiadamente pequenos geram o problema de deixar o acesso a esses arquivos mais lento, considerando que um grande arquivo necessita de vários blocos e estes vários locais de acesso.

O que entra em questão nesse momento então é definir qual o tamanho ideal para o bloco. Estudos realizados por Tanenbaum em 1984 e em 2005, que são citados no livro Sistemas Operacionais Modernos, chegaram à conclusão que o tamanho ideal para o bloco é de 4KB, porém podem ser implementados sistemas de arquivos com diferentes tamanhos em determinadas situações. A tabela abaixo descreve o estudo de forma mais clara, mostrando o percentual de arquivos que tem tamanho menor que o tamanho em questão.

Tamanho	UV 1984	UV 2005	Web
1	1,79	1,38	6,67
2	1,88	1,53	7,67
4	2,01	1,65	8,33
8	2,31	1,80	11,30
16	3,32	2,15	11,46
32	5,13	3,15	12,33
64	8,71	4,98	26,10
128	14,73	8,03	28,49
256	23,09	13,29	32,10
512	34,44	20,62	39,94
1 KB	48,05	30,91	47,82
2 KB	60,87	46,09	59,44
4 KB	75,31	59,13	70,64
8 KB	84,97	69,96	79,69

Tamanho	UV 1984	UV 2005	Web
16 KB	92,53	78,92	86,79
32 KB	97,21	85,87	91,65
64 KB	99,18	90,84	94,80
128 KB	99,84	93,73	96,93
256 KB	99,96	96,12	98,48
512 KB	100,00	97,73	98,99
1 MB	100,00	98,87	99,62
2 MB	100,00	99,44	99,80
4 MB	100,00	99,71	99,87
8 MB	100,00	99,86	99,94
16 MB	100,00	99,94	99,97
32 MB	100,00	99,97	99,99
64 MB	100,00	99,99	99,99
128 MB	100,00	99,99	100,00

Dentre as estruturas de dados utilizadas no armazenamento, podemos citar algumas:

Mapa de bits: As entradas da tabela são associadas a cada bloco do disco e esta por sua vez é representada por um bit que é usado como uma *flag*, ou seja, quando estiver sinalizando 1(*true*) indicará que o bloco do disco está sendo usado e quando essa *flag* indicar 0 (*false*) representa que o bloco em questão está com o seu espaço livre para armazenamento.

Considerações finais

Referências:

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

<http://wiki.inf.ufpr.br/maziero/lib/exe/fetch.php?media=so:filesystem.pdf>

<http://docente.ifsc.edu.br/alex.forghieri/MaterialDidatico/Sistemas%20Operacionais/Material%20das%20aulas/06%20-%202024-06-2016/Sistema%20De%20Arquivos.pdf>