

# Sistemas Operacionais

Prof. Me. Pietro M. de Oliveira

## Unidade III

Conceitos e tipos de memória.

Endereçamento físico e lógico.

Fragmentação.

Estratégias de alocação de memória.

Memória virtual.

Conceito de arquivo.

Sistemas de arquivos.

## Processos:

Integralmente ou parcialmente em memória.

## CPU × Memória

Frequência de funcionamento.

Acesso direto.

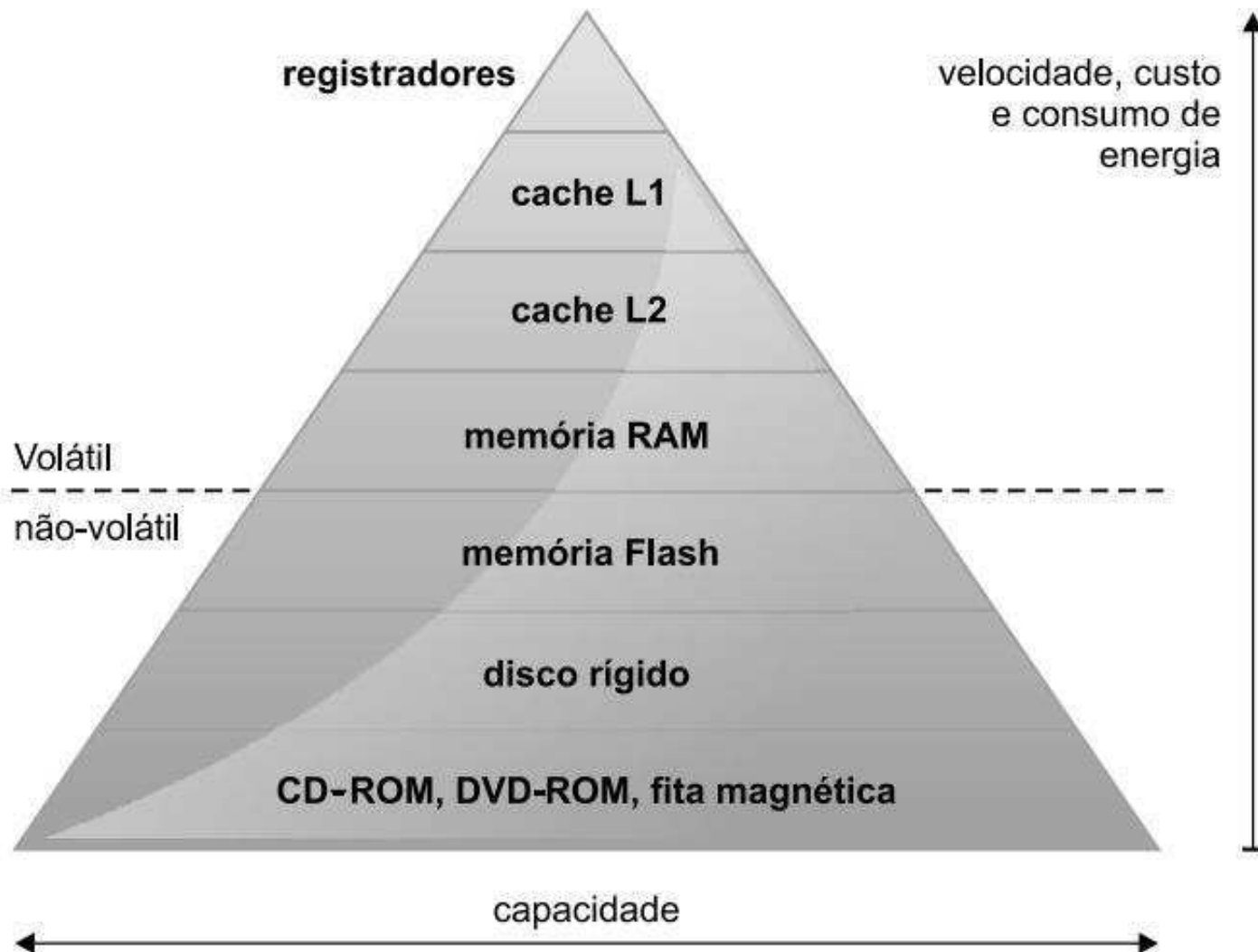
## Níveis de Memória

Embutida.

Primária (Principal).

Secundária (Externa).

# Hierarquia de Memória



Programa de alto nível (C++, Java, C#)

Programador não se preocupa c/ end.

Variáveis relacionam-se c/ end. Lógico.

Endereço lógico × endereço físico

São diferentes, geralmente.

*Memory-Management Unit* (MMU).

SO:

Alocar endereços lógicos aos programas de usuário.

## Funcionamento da MMU



Fonte: Wolts, V. (2016, p. 66)

Processos entram e saem de execução:

Memória é alocada e desalocada.

Fragmentação consiste em 'buracos' de memória não utilizados.

Dois tipos:

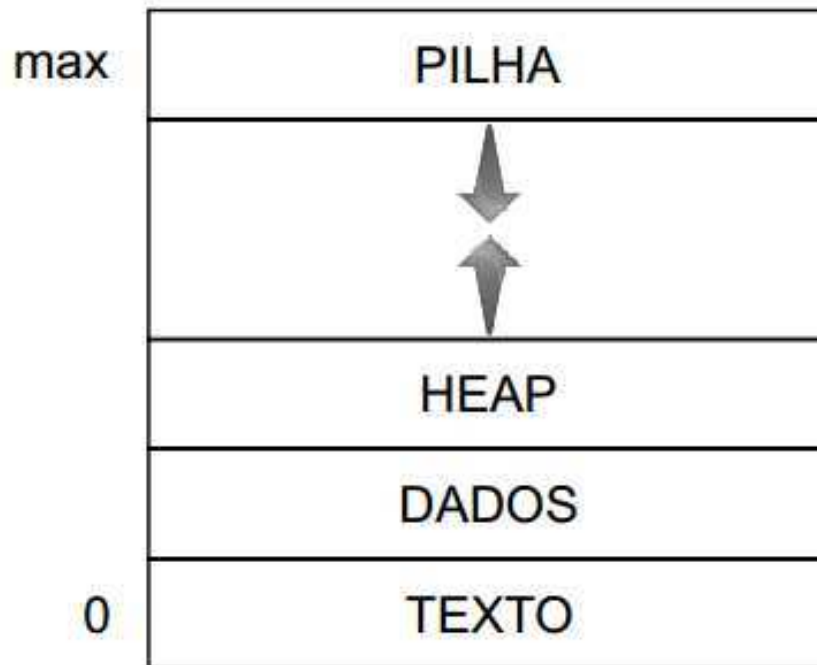
Externa e interna.

Exemplo:

Memória de 10 posições.

4 processos: A, B, C e D.

## Exemplos de fragmentação interna.



Processo em memória

Fonte: Silberschatz (2011, p.57)



## Exemplos de fragmentação externa

1	2	3	4	5	6	7	8	9	10

1	2	3	4	5	6	7	8	9	10
A	A	A							

1	2	3	4	5	6	7	8	9	10
A	A	A	B	B					

1	2	3	4	5	6	7	8	9	10
A	A	A	B	B					

1	2	3	4	5	6	7	8	9	10
			B	B					

1	2	3	4	5	6	7	8	9	10
			B	B	C	C	C	C	C

1	2	3	4	5	6	7	8	9	10

1	2	3	4	5	6	7	8	9	10
A	A								

1	2	3	4	5	6	7	8	9	10
A	A	B	B	B					

1	2	3	4	5	6	7	8	9	10
A	A	B	B	B	C	C	C	C	C

1	2	3	4	5	6	7	8	9	10
A	A	B	B	B	C	C	C	C	C

1	2	3	4	5	6	7	8	9	10
A	A	D	D		C	C	C	C	C

## Memória principal (RAM)

Gerenciada pelo SO.

Armazena dados de processos.  
de Usuário ou do próprio SO.

## Estratégias:

Alocação contígua.

Alocação por segmentos.

Alocação paginada.

Memória fracionada em duas partições:

- Primeira reservada ao SO.

- Segunda para os processos de usuário.

  - Foco nas demandas específicas.

  - Tamanhos ajustáveis.

Desenvolvimento simples:

- SO solicita um end. lógico à MMU.

- Caso não haja espaço: interrupção.

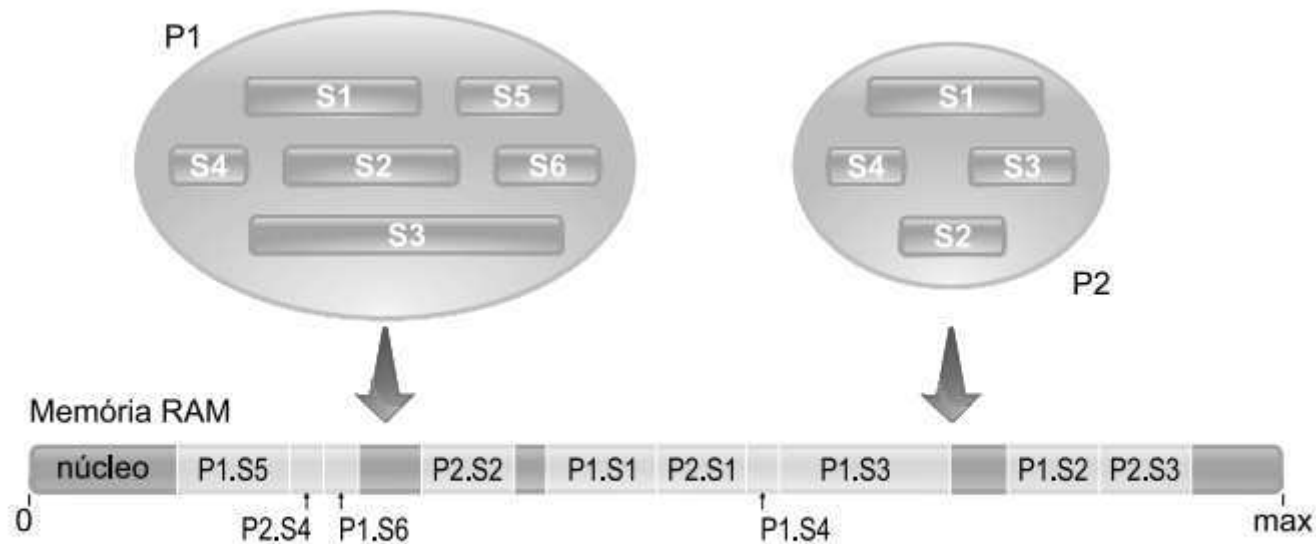
Está sujeita a fragmentação externa .

Memória fracionada em segmentos:

Processos ficam segmentados na mem.

Segmentos de tamanhos variados.

Sujeita a fragmentação externa



Fonte: Maziero (2011, cap. 5, p.13).

Disponível em: <<http://dainf.ct.utfpr.edu.br/~maziero/lib/exe/fetch.php/so:so-cap05.pdf>>

Endereço lógico dividido em blocos

Páginas (lógico) alocada a quadros (físico).

Programa não 'enxerga' páginas (transparência).

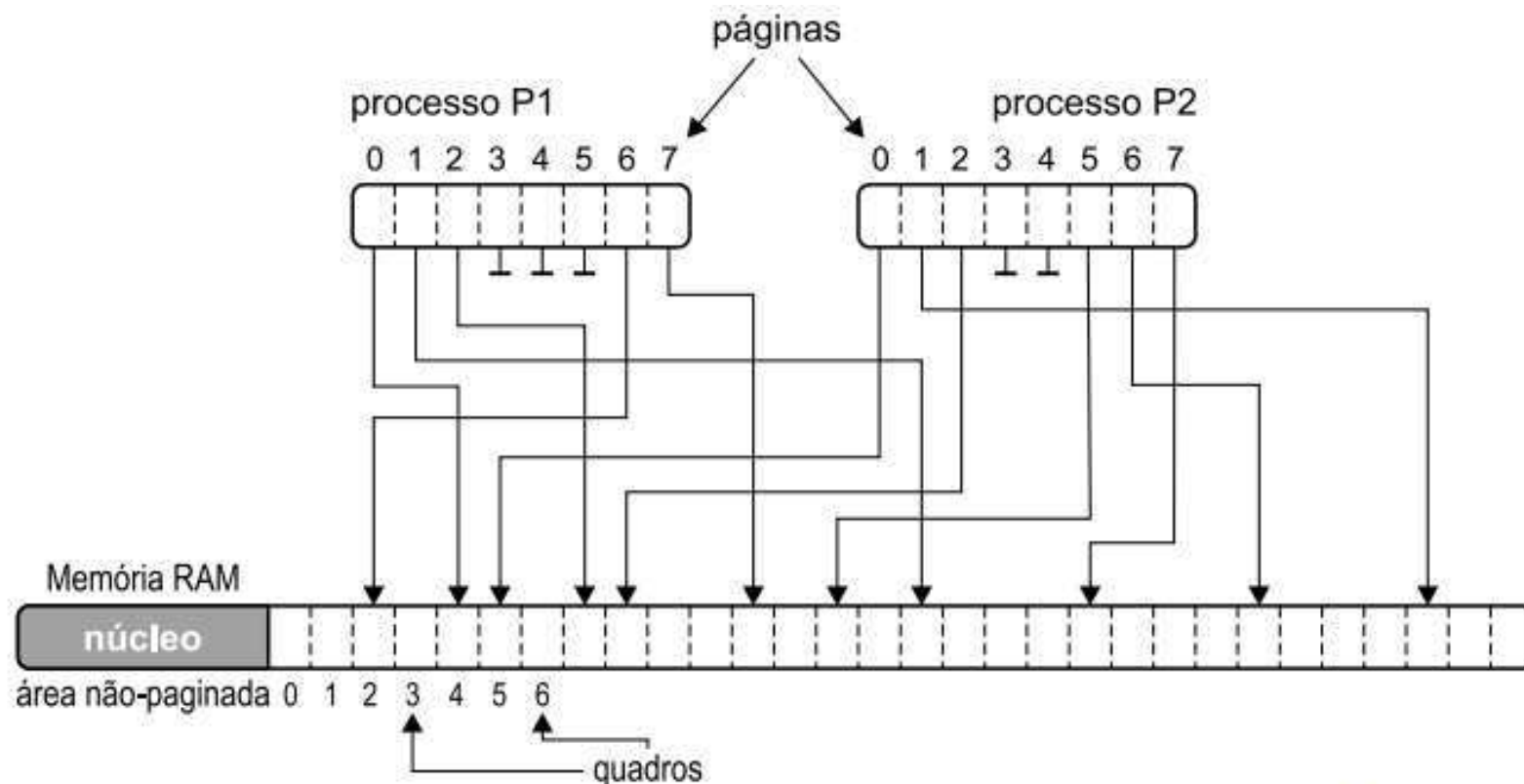
Auxilia na compactação de dados.

Evita fragmentação externa.

Tabela de páginas

Associação página-quadro.

Implementada na maioria dos SOs atuais.



Fonte: Maziero (2011, cap. 5, p. 16). Disponível em: <<http://dainf.ct.utfpr.edu.br/~maziero/lib/exe/fetch.php/so:so-cap05.pdf>>

## Problema:

E se a memória estiver cheia?

Como proceder com dados de processos que não podem ser desalocados?

E os processos que exigem ser carregados por completo na memória?

## Solução:

Mais memória!



Memória virtual utiliza memória secundária (externa), como se fosse memória principal

HD passa a ser uma extensão da RAM.

*Swapping:*

Algoritmo escolhe página p/ sair da RAM.

Memória principal recupera espaço.

Transparente para o usuário (programador).



*First In, First Out* (FIFO):

Páginas mais antigas são retiradas primeiro.

*Least Recently Used* (LRU):

Páginas menos usadas recentemente são retiradas primeiro.

*Not Recently Used* (NRU):

As páginas que não foram usadas recentemente sairão por primeiro.

Prioridade de processos ou conteúdo de páginas também são critérios válidos.

Para a maioria dos usuários, o Sistema de Arquivos é o aspecto mais visível do SO.

Responsável por:

- Armazenar dados.

- Prover acesso aos dados.

- Programas do SO.

- Programas do Usuário.

## Memória principal:

- Muito pequena.

- Volátil.

- Dados de processos podem ser volumosos.

## Necessidade de persistência

- Dados armazenados permanentemente.

- Grandes quantidades de dados.

## Memória secundária:

- DVDs, Pen Drives, HD.

Disco Rígido (HD):

Conjunto de discos magnéticos de alta rotação.

Braço + cabeçote.

Setores: blocos de tamanho fixo.

Partições: região com um conjunto de setores

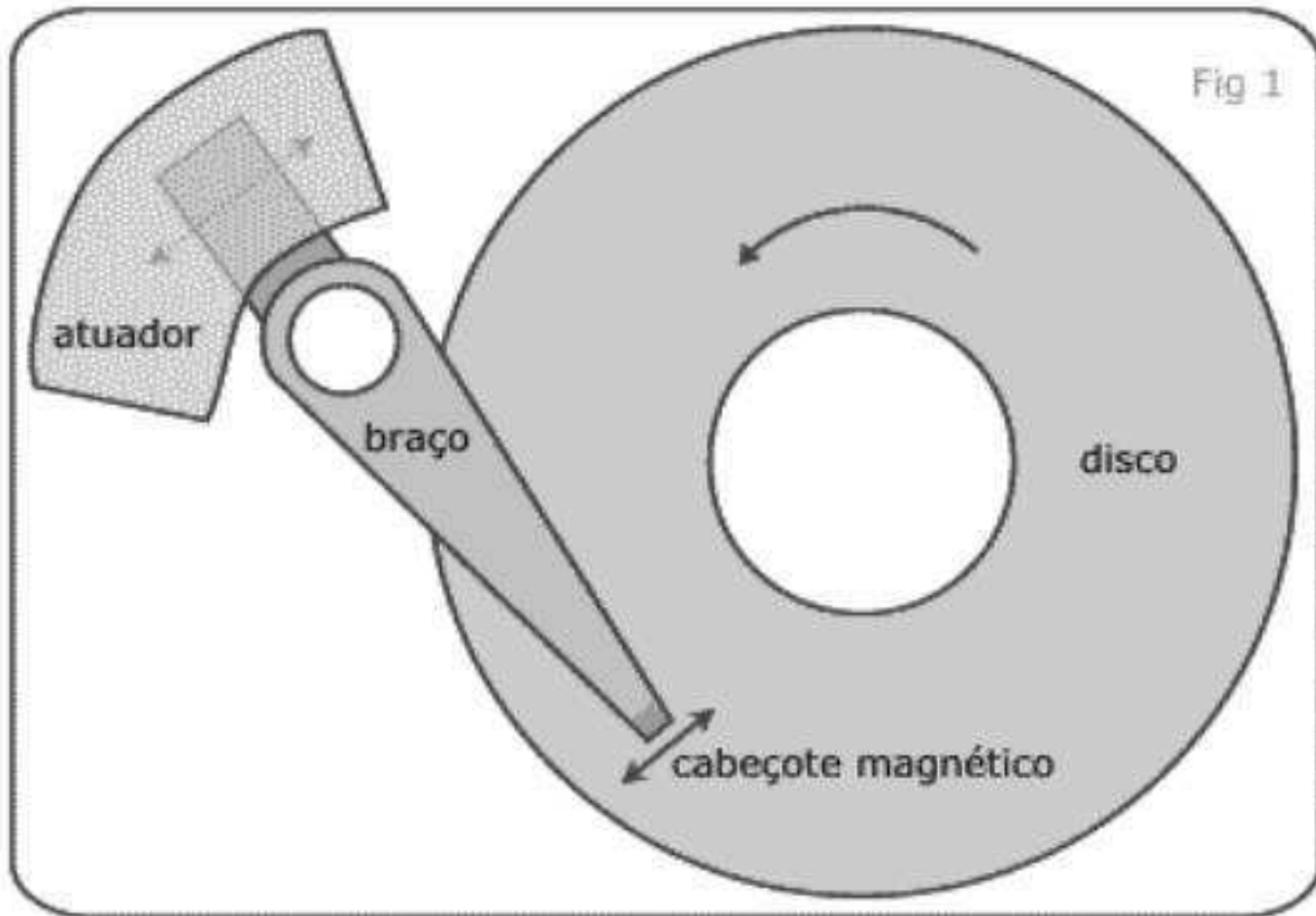
SO é, geralmente, instalado aqui.

Pode-se ter **N** partições com **N** sistemas operacionais

Partição especial: *Master Boot Record* (MBR)

Primeira partição a ser acessada.

Formatação: estabelece um sistema de arquivos.





Fonte: sxc / freepik.com



Arquivos são conjuntos de dados armazenados em um dispositivo não volátil.

Atributos:

- Nome.

- Extensão ou tipo.

- Data.

- Tamanho.

- Proprietário.

- Permissões.

- Senha.

O SO não reconhece todos os tipos de arquivos.

Arquivos específicos – programas específicos.

O SO ‘se vira’ com um número alto de arquivos.

Ser humano: necessita organização lógica.

Diretórios são facilitadores para organizar e hierarquizar os arquivos.

Auxilia localização e pesquisa.



Um sistema de arquivos organiza física e logicamente os arquivos e diretórios.

Formatação: organização física da mem. Secundária.

Instalação do SO: organização lógica.

Baseado nas estratégias de alocação de arquivos.

Sistemas de arquivos mais comuns:

- NTFS – Windows XP, Vista, 7 e 8.

- Ext2, Ext3 ou Ext4 – Linux.

- HFS+ – Mac OS.

- FAT16 e FAT32 – Windows e Pen Drives.

## Alocação contígua

Armazena os blocos lado a lado.

Fácil implementação.

Endereço do arq. definido no primeiro bloco.

Acesso e pesquisa rápidos.

Dados aglutinados.

Leitura em um único acesso sequencial.

Sujeito a fragmentação externa.

Solução paliativa: desfragmentador.

## Alocação encadeada

Blocos são alocados aonde for possível.

Cada bloco aponta para o próximo bloco.

Evita fragmentação externa.

Arquivos ficam espalhados no disco.

Perda de desempenho.

Desfragmentador 'aproxima' blocos.

*File allocation table* (FAT) – Surgiu com o tempo

Guarda as posições dos próx. Blocos.

Desvantagem: pode tomar muito espaço!

## Alocação indexada ou *i-nodes*

Estrutura de dados: nó-índice.

Relaciona atributos e endereços dos blocos.

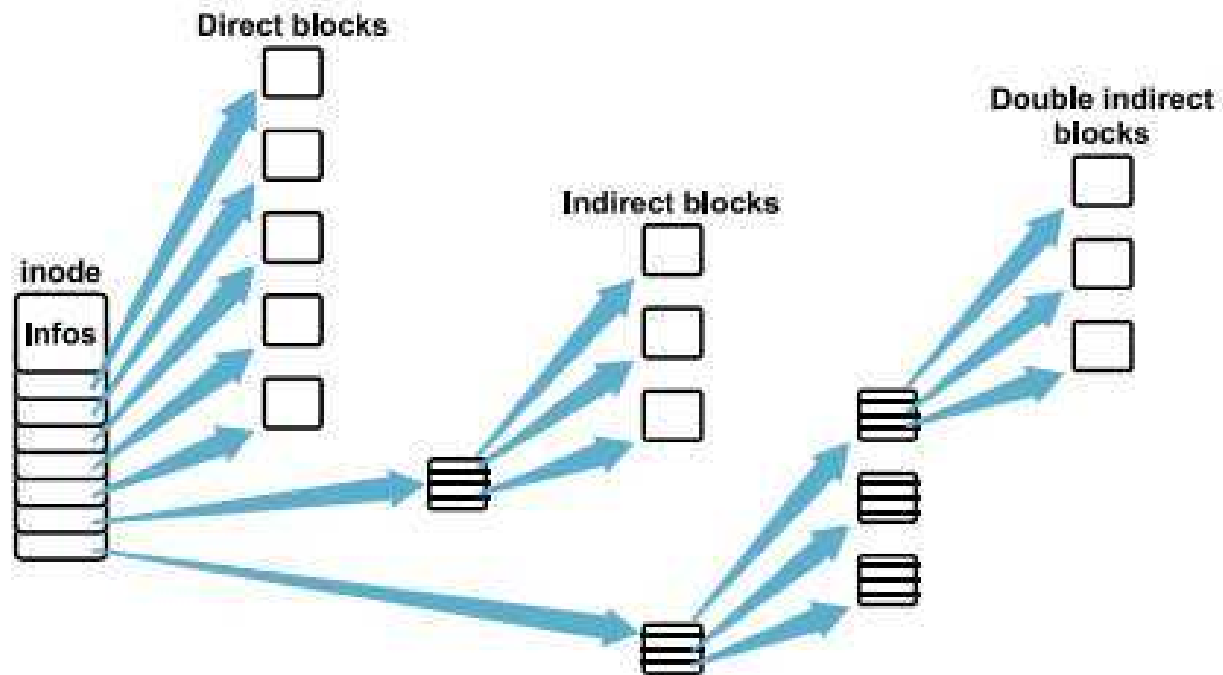
Não há necessidade da FAT.

FAT pode tomar muito espaço.

Vantagem: carrega-se o nó-índice somente se o arquivo estiver aberto.

Utilizada normalmente no Linux.

# Técnicas para Alocação de Arquivos



**Alocação indexada**

Fonte: <<http://upload.wikimedia.org/wikipedia/commons/a/a2/Ext2-inode.gif>>

Hierarquia de memória.

Endereçamento lógico e físico.

Fragmentação interna e externa.

Alocação de memória.

Contígua, por setores e paginada.

Memória virtual.

Técnicas de *swapping*.

Sistemas de arquivos.

HD, definição de arquivos e diretórios.

Técnicas para alocação de arquivos.

# Sistemas Operacionais

Prof. Me. Pietro M. de Oliveira