

**Ministério da Educação**  
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca  
UNED Nova Friburgo  
Curso Técnico em Informática Integrado ao Ensino Médio

# **Memória primária e secundária**

## **Sistemas Operacionais**

Prof. Bruno Policarpo Toledo Freitas  
13 de maio de 2020  
[bruno.freitas@cefet-rj.br](mailto:bruno.freitas@cefet-rj.br)

# Objetivos

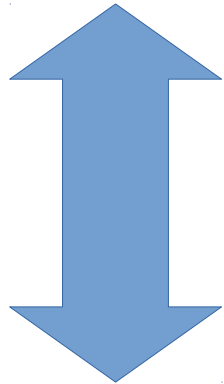
- **Entender o princípio de funcionamento dos sistemas de arquivos**
- **Apresentar os principais sistemas de arquivos**
- **Entender as diferenças entre os sistemas de arquivos**
- **Configurar sistemas de arquivos e partições no GNU/Linux**

# Introdução

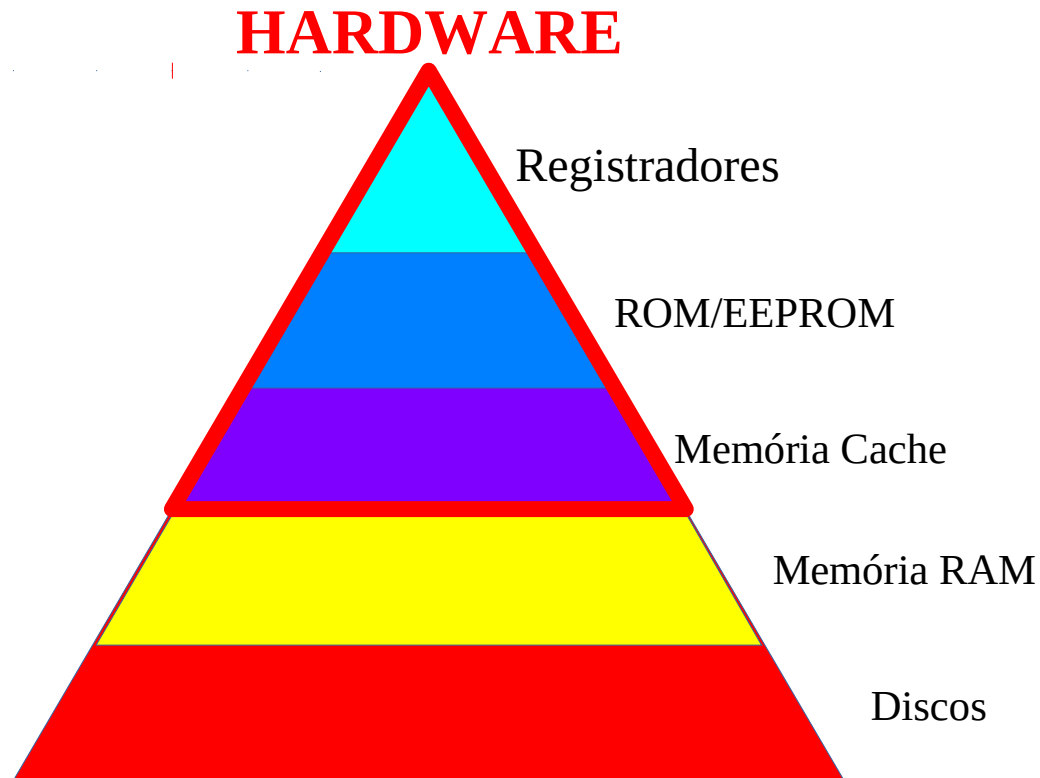
- **A memória é dividida em 2 tipos: a primária e a secundária**
  - Primária: RAM
  - Secundária: Armazenamento
- **O sistema operacional junto com o hardware realiza diversas enxerga toda essa memória como se fosse uma só**
  - Memória virtual

# Hierarquia de memória

Custo maior  
Velocidade maior  
Capacidade menor



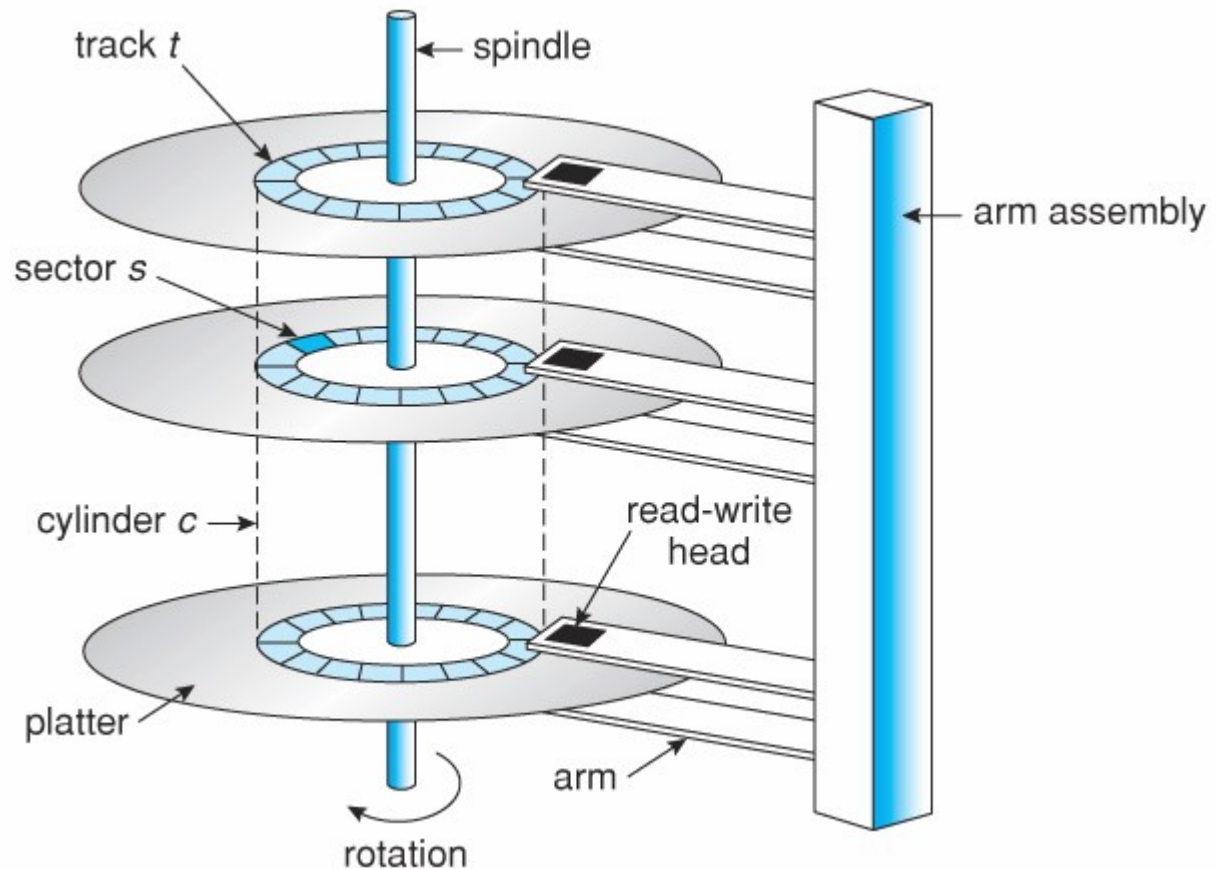
Custo menor  
Velocidade menor  
Capacidade maior



# Memória primária $\leftrightarrow$ Memória secundária

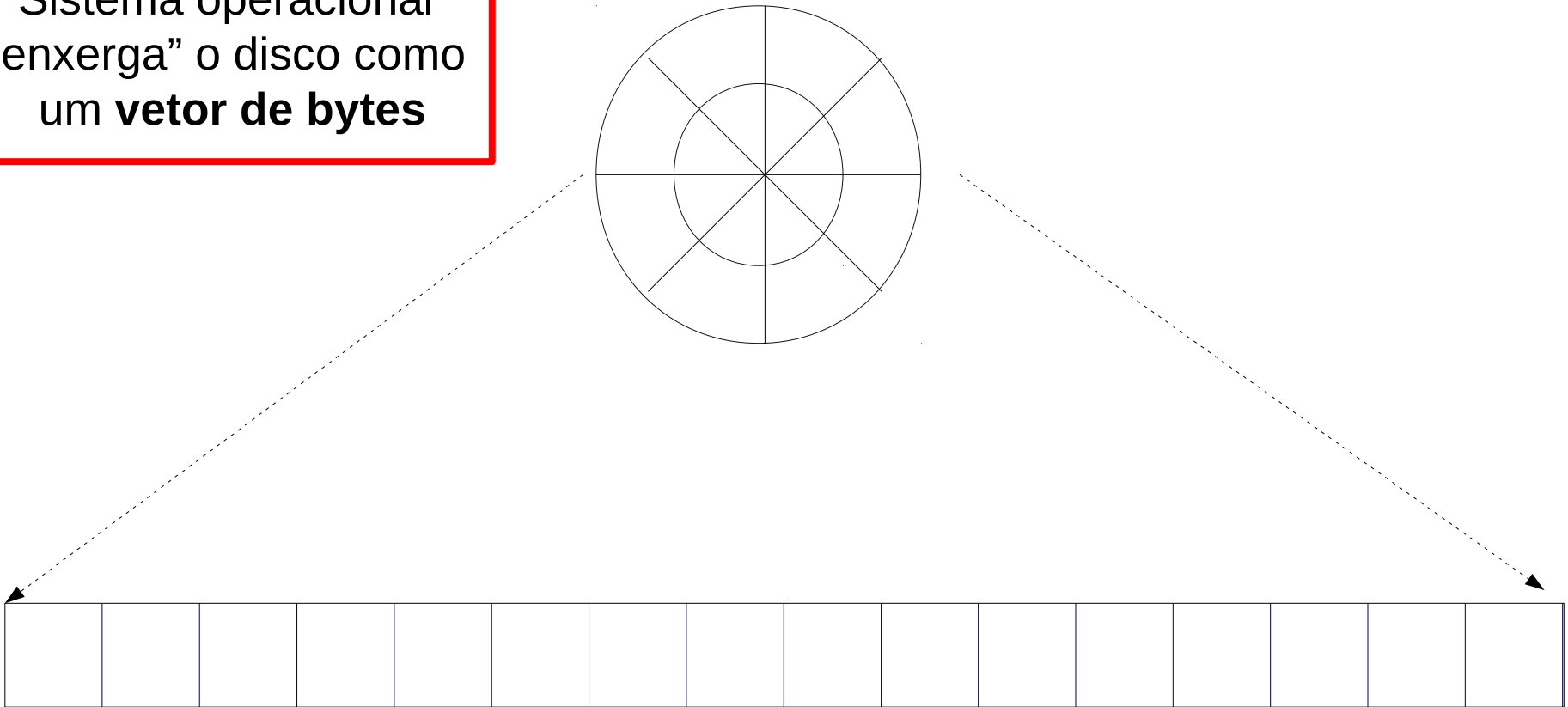
- **Nós não controlamos o uso da memória primária diretamente**
  - Hardware + Sistema Operacional
- **O que podemos controlar são alguns parâmetros**
  - Se iremos utilizá-la ou não (usual: ligada)
  - Tamanho da área de swap
- **Windows: Arquivo de paginação**
- **Linux: partição de swap**

# Estrutura física do discos magnético



# Estrutura física de um arquivo

Sistema operacional  
“enxerga” o disco como  
um **vetor de bytes**

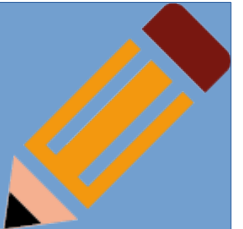


# Arquivo

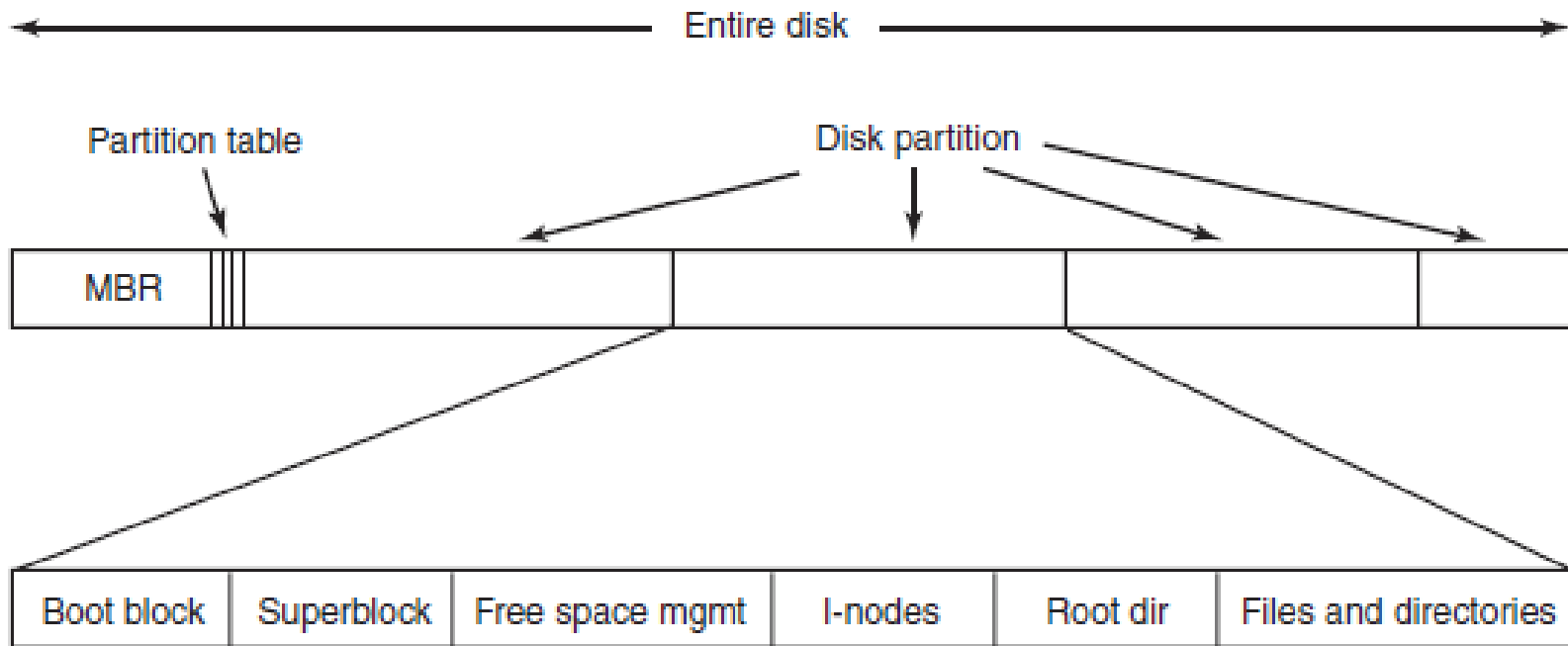
- **O arquivo é uma entidade de armazenamento composta por:**
  - Blocos: onde são armazenados os dados em si
    - Tamanho do bloco definido na formatação da partição
      - 512B, 1KB, 2KB, **4KB**, 8KB, 16KB, 64KB ...
  - Índices (i-nodes): Lista de blocos que compõem o arquivo
  - Metadados: Informações sobre o arquivos
    - Nome, permissões, dono e grupo, MAC times (Modificado-Acessado-Criado), ...



Partições



# Layout geral de sistemas de arquivo



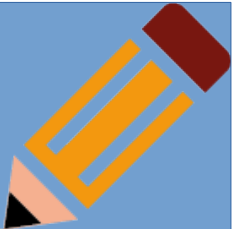
# Partições & Tabela de partições

- **O primeiro setor do disco (512B) é o setor de boot**
- **Normalmente ele contém o bootloader e uma tabela de partições**
- **A tabela de partições do tipo MBR (desuso) contém:**
  - Até 4 partições primárias
  - Uma partição de boot
- **Partições GPT permitem tamanho de sistemas de arquivos maiores**

# Editor de partições

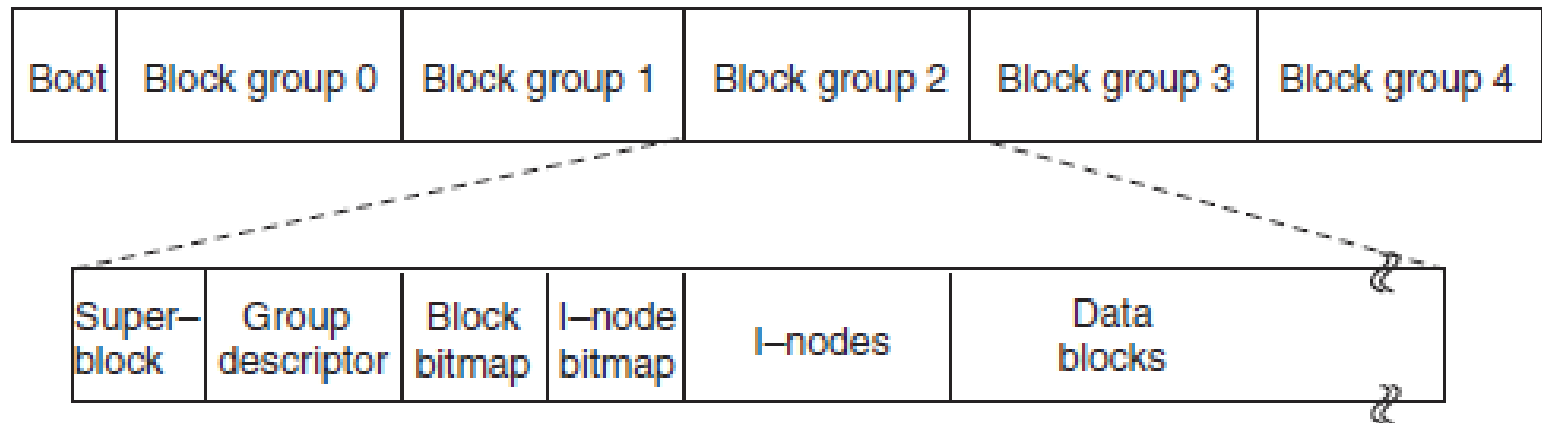
- **Aplicação GNU/Linux para editar partições, tabelas de partições, e informar sobre a geometria do disco**
- **Utilização:**  
`[sudo] fdisk [disco]`
- **Usado para fins de restauração da tabela de partições do disco**
  - **Vocês podem potencialmente ferrar a tabela de partições se vocês a usarem de forma errada!**

# Sistemas de arquivos



# Sistema de arquivo GNU/Linux ext2

- **Extended Filesystem 2**
- **Performance superior ao ext e MINIX 1**
- **Tamanho total máximo do sistema de arquivos de 64 TB**
- **Estrutura do sistema de arquivos:**



# Sistemas de arquivos GNU/Linux

## ext3/ext4

- **Problema do ext2: confiabilidade**
  - Se der der pane no sistema durante uma operação de escrita, o sistema de arquivos pode ficar inconsistente
- **ext3: adição da jornalização**
  - Mantém um jornal das operações que serão escritas no disco
  - Se houver crash, o sistema desfaz operações incompletas
- **ext4: suporte a jornalização em arquivos e sistemas de arquivos maiores**

# Sistemas de arquivo Windows

## FAT16/32

- **Usada no MS-DOS e Windows 1.0 até Me**
- **Utiliza uma tabela na memória pra organizar os blocos dos arquivos**
- **FAT32: Arquivos de 2TB**
- **Pouca complexidade**
- **Problema de escalabilidade**



# Sistemas de arquivo Windows

## NTFS

- **Introduzido no Windows XP**
- **Arquivos de até (teoricamente)  $2^{64}$  bytes**
- **Nome de arquivos e diretórios de 255 caracteres**
- **Case-insensitive**
- **Suporte a journalização**

# Limites dos sistemas de arquivos

<b>Tipo</b>	<b>Tamanho máximo dos nomes de arquivos</b>	<b>Tamanho máximo do arquivo</b>	<b>Tamanho máximo da partição</b>
ext2 (bloco com 4KB)	255	2TB	16TB
ext3 (bloco com 4KB)	255	2TB	16TB
ext4	256	16TB	1EB
FAT32	255	4GB	2TB
NTFS	255	16EB	16EB

# Gerência de Sistemas de Arquivos



# Aplicações de gerência de sistema de arquivos

- **Windows 7 / Windows 10:**

- Painel de controle → Sistema e Segurança → Ferramentas Administrativas → Gerenciamento do computador

- **GNU/Linux:**

- fdisk: gerenciador de tabelas de partição
- parted / gparted: gerenciador de sistemas de arquivos e partições

# Visão geral

- **O hardware de armazenamento aparece na /dev:**
  - /dev/sdaX : discos magnéticos e flash
  - /dev/cdrom: discos ópticos
- **Para disponibilizar os sistemas de arquivos no Linux é necessário:**
  - Saber o tipo do sistema de arquivos
  - Montá-los no diretório raiz

# mount

- **Monta o sistema de arquivos em uma determinada pasta**
- **Utilização padrão:**
  - [sudo] mount *dispositivo* [-t *tipo*] [-o *opções*] *pasta*
  - Opções são separadas por vírgulas
- **Parâmetros:**
  - mount -l : lista os sistemas de arquivos montados
  - mount -a : monta todos os sistemas do arquivo /etc/fstab

# mount

## Tabela de opções genéricas

Opção	Descrição
auto	Monta automaticamente com o mount -a
async/sync	Escritas assíncronas/síncronas no disco ( <u>nunca usar</u> )
defaults	Monta com opções padrão: rw, suid, dev, exec, auto, nouser, async
dev/nodev	Interpreta arquivos especiais
exec/noexec	Permite a execução de programas pelo sistema de arquivos
group	Permite que um usuário monte o sistema de arquivos se ele pertencer ao mesmo grupo do dispositivo
rw, ro	Monta o sistema de arquivos escrita/leitura ou somente leitura
suid/nosuid	Bits de u/g/o possuem/não possuem efeito
user/nouser	Sistema de arquivos pode/não pode ser montado por usuários (sem sudo)

# umount

- **Desmonta um sistema de arquivos previamente montado**
- **Utilização:**
  - [sudo] umount [pasta ou dispositivo]



# gparted

- Aplicação gráfica para editar partições do disco
- **[sudo] apt-get install gparted**
- Sem ela, é necessário usar o *mkfs* junto com o *fdisk*

# /etc/fstab

- Arquivo que descreve os sistemas de arquivos estáticos do Linux.

```
bruno@bruno-VirtualBox:~$ cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>    <type>  <options>          <dump>  <pass>
# / was on /dev/sda1 during installation
UUID=884b182e-aac9-4cfa-ad18-55d42f4813b7 /        ext4      errors=remount-ro 0        1
# /home was on /dev/sdb1 during installation
UUID=950fedf6-7a6f-4f9b-9b37-7b0f1b97e40d /home    ext4      defaults          0        2
# /var was on /dev/sdc1 during installation
UUID=449de7c2-acaa-47dc-af71-9772f341d2b3 /var     ext4      defaults          0        2
/swapfile                                none      swap            0        0
# Dropbox folder
Dropbox /home/bruno/Dropbox    vboxsf    defaults          0        2
```

# mkfs

- **Cria sistemas de arquivos em uma partição.**
- **Cada sistema de arquivos possui uma versão do mkfs para si**

mkfs.ext4, mkfs.ntfs, mkfs.ext2 ...

- **Utilização:**

[sudo] mkfs.[tipo] [partição]

# Exercício 1

- Qual é o tamanho de um disco com 256 setores por trilha, 25 trilhas, com 4 cabeçotes, e 512 bytes por setor?

# Exercício 1

**Utilizando o *fdisk*, descubra:**

- a) Quantos blocos o disco possui
- b) O tamanho do bloco do disco

**Agora, multiplique o tamanho do bloco pela quantidade de blocos do disco. O tamanho é igual ao informado pelo sistema? Se não, explique o porquê.**

## Exercício 2

**Utilizando o *gparted*, crie uma partição de 1GB tomando os seguintes passos:**

- a) Utilizando o *gparted*, desmonte alguma partição e diminua seu tamanho em 1GB
- b) Utilizando o *gparted*, crie a partição espaço de 1GB com o espaço liberado pelo item anterior
- c) Crie um sistema de arquivos *ext4* na partição criada
- d) Teste a nova partição: monte e desmonte a partição criada em alguma pasta na *home* do usuário
- e) Edite o arquivo */etc/fstab* a fim de que a nova partição seja montada no sistema de arquivos raiz sempre que o computador iniciar

## Exercício 3

**Instale uma distribuição GNU/Linux novamente, porém utilizando o criador avançado de partições com as seguintes partições:**

/home : 100GB

/var: 5GB

/ : O restante do espaço

**Agora, verifique o arquivo /etc/fstab.**

# Desafio!

**Repita o exercício 2, mas utilizando o *fdisk* junto com o aplicativo *mkfs* (ou seja, sem interface gráfica). Tome os seguintes passos:**

- Crie um ponto de restauração da máquina virtual (você vai precisar)
- Diminua o tamanho de alguma partição do sistema usando *mkfs*
- Altere os blocos de início e fim das partições usando o *fdisk*
- Crie um sistema de arquivos no espaço livre utilizado usando o *mkfs*
- Monte o novo sistema de arquivos utilizando o *mount*.



# Referências

- FILHO, João Eriberto Mota. **Descobrimdo o Linux: entenda o sistema operacional GNU/Linux**. 3ª. ed. São Paulo: Novatec Editora, 2012.
  - Cap 5.: seções 5.1, 5.2, 5.5, 5.8, 5.9, 5.17