

Capítulo 7

Gerenciamento de arquivos e memórias

- Estrutura de diretórios do Windows
- Estrutura de diretórios do Linux
- Prompt de comandos do Windows
- Prompt de comandos do Linux
- Sistemas de arquivos
- Montagem de dispositivos no Linux

Toda vez que executamos um utilitário ou um aplicativo no computador, estamos indiretamente manipulando arquivos e dados na memória. Dessa forma, também em outras ocasiões precisamos criar, copiar ou transferir arquivos de um lugar para outro do disco rígido (ou entre dispositivos de armazenamento). É muito importante dominar diretórios e pastas em um Sistema Operacional, sabendo em que local (fisicamente) está armazenado o arquivo que acabamos de criar, para que se possa, entre outras iniciativas, movê-lo para outro dispositivo.

7.1. Estrutura de diretórios do Windows

Quando o Windows é instalado em um computador, automaticamente cria uma estrutura de pastas (também chamados diretórios), onde são organizados seus arquivos (tabela 1). Assim, a tarefa de encontrar determinado arquivo se torna mais fácil, pois ele está sempre armazenado em pastas predeterminadas.

Tabela 1
Estrutura de pastas do Windows.

Diretório	Conteúdo
C:\windows\fonts	Arquivos de fontes (tipos de letras que serão utilizados pelos softwares, como o Word, por exemplo).
C:\windows	Arquivos e pastas do sistema operacional Windows.
C:\arquivos de programas	Arquivos e pastas de aplicativos que foram instalados pelo usuário ou administrador do computador.
C:\documents and settings (windows xp)	Arquivos e pastas pessoais dos usuários que estão cadastrados no computador (inclui a pasta Meus Documentos e Desktop de cada usuário).
C:\usuarios (windows vista)	
C:\windows\system	Pastas do sistema: contém programas executáveis, arquivos e pastas importantes para o sistema operacional.
C:\windows\system32	
C:\windows\system32\drivers	

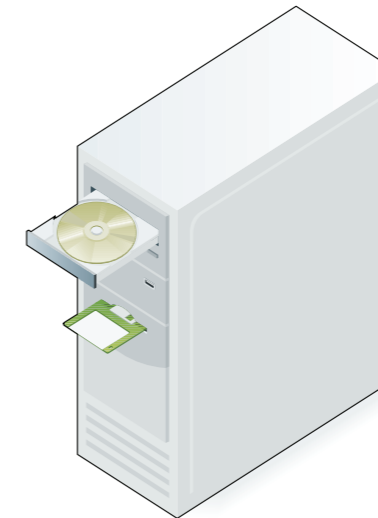


Figura 176
Imagem das unidades de disco.

No início de todos os diretórios que aparecem na tabela 1 temos a letra “C:”, que representa a unidade de disco utilizada pelo Sistema Operacional. Você pode até ter disponível em seu PC mais de uma unidade de disco (D:, E: etc.) e o Windows pode ter sido instalado em outra unidade de disco, como a D:, por exemplo (figura 176). Para visualizar as unidades de disco disponíveis em seu PC, clique no menu “Iniciar” e depois em “Computador”:

7.2. Estrutura de diretórios do Linux

No Linux, assim como no Windows e em qualquer outro sistema operacional, há uma organização de pastas que visa melhorar o desempenho do sistema e facilitar a vida do administrador na hora de corrigir algum problema, instalar um novo programa etc. A organização de diretórios no Linux é a que aparece na tabela 2.

Tabela 2
Estrutura de diretórios do Linux.

Diretório	Conteúdo
/home	Pastas pessoais dos usuários comuns
/root	Pasta pessoal do usuário administrador
/etc	Arquivos de configuração
/dev	Apontadores para dispositivos (arquivos que simbolizam dispositivos)
/var	Bibliotecas e registros do sistema
/bin	Arquivos binários: comandos do sistema
/boot	Arquivos de inicialização
/lib	Bibliotecas do sistema
/media ou /mnt	Diretório de montagem de dispositivos
/opt	Programas que não são nativos da distribuição
/var	Arquivos de registro (log)

No Linux, não existem unidades de disco, e sim diretórios de montagem de dispositivos, ou seja, durante a inicialização do sistema, a partição utilizada pelo sistema operacional é montada na pasta "/". É possível fazer uma analogia com o Windows, já que neste o diretório raiz (o primeiro disponível no disco) é o "C:\", e no Linux existe somente "/". Se adicionarmos um novo disco rígido ou uma nova partição de disco, isso terá de ser montado em uma pasta qualquer no sistema operacional, como a home, por exemplo. Assim, quando salvarmos um arquivo nessa pasta, ele estará sendo salvo fisicamente no segundo dispositivo ou partição do HD.

Nesse caso, não existe uma letra para representar a unidade de disco utilizada pelo sistema operacional. Outra característica importante é que, no Linux, as pastas são separadas por uma barra normal "/", enquanto no Windows é usada a invertida "\".

7.3. Prompt de comandos do Windows

Já sabemos que o sistema operacional possui uma linguagem de comandos disponíveis para manipulação das diversas tarefas que podem ser executadas pelo sistema. É possível gerenciar boa parte do Windows pelo prompt, ou seja, podemos executar tarefas diversas, desde as mais básicas, como criar uma nova pasta, até as mais avançadas, como alterar as configurações de rede do PC. O prompt de comandos do Windows é um aplicativo em que o usuário dispõe de uma interface que interpreta os comandos digitados. Estes são transferidos ao sistema operacional, que, depois de executá-los, retorna o resultado ao prompt, de maneira que o usuário o visualize.

7.3.1. Introdução

Para executar o prompt de comandos do Windows, clique no menu "Iniciar" e depois digite o comando "cmd" na caixa "Iniciar Pesquisa". Outra forma de executar o prompt de comandos do Windows é pelo submenu "Acessórios" (figura 177).

Quando executamos o prompt de comandos, a primeira informação que recebemos é sobre a versão do Windows. Aparecerá um cursor piscando, para indicar que o prompt está pronto para receber comandos (figura 178).

Para executar um comando no prompt, basta digitá-lo e pressionar, no final, a tecla "Enter" do teclado. Vamos testar nosso primeiro comando no prompt utilizando o comando "dir", que serve para visualizar o conteúdo do diretório atual. À esquerda

Figura 177

Abrindo o prompt de comandos do Windows.

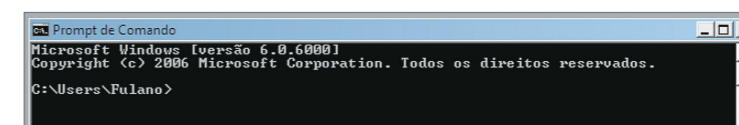
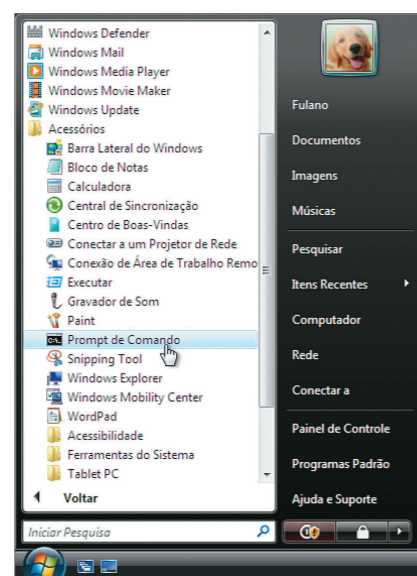


Figura 178

Prompt de comando.

do cursor, aparece o caminho para o diretório atual, o que nos leva a concluir que o prompt vai exibir os arquivos e as pastas disponíveis no diretório "Fulano", que é uma subpasta do diretório "Users" (usuários), da unidade C (figura 179).

O retorno do comando é uma listagem de pastas e diretórios contidos na pasta "Fulano", que é o nome do usuário que utilizamos para entrar no sistema.

Na primeira linha temos a informação de que a unidade C não tem nome. Isso significa que podemos dar-lhe um nome, seja clicando com o botão direito do mouse sobre ela, na janela "Computador", seja via mando "label", seguido do nome que será dado à unidade (figura 180).

Você perceberá, também, a existência de um código (número de série) que representa a unidade C quando digitamos o comando "dir". Esse número é dado pelo próprio sistema operacional, quando formatamos a unidade, e serve para diferenciar a unidade atual de outras disponíveis na máquina. Deve, portanto, ser único, como o nome da unidade. Nas colunas que surgem em seguida, temos a data e a hora da criação do arquivo ou pasta, um símbolo "<DIR>", que indica tratar-se de um diretório (ou pasta), e não de um arquivo e, por fim, o nome do arquivo ou pasta.

Figura 179

Comando "dir".

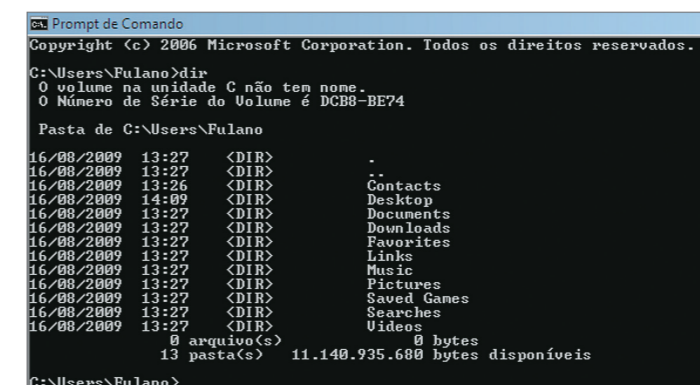


Figura 180

Renomeando a unidade de disco.

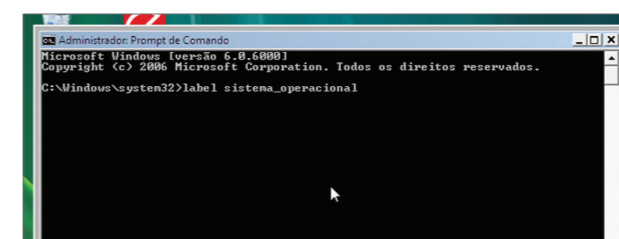
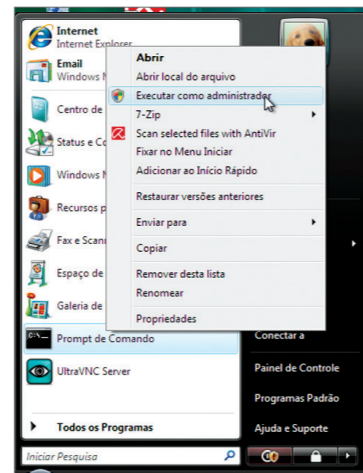


Figura 181

Executando o prompt como administrador.



IMPORTANTE
 Não pode haver dois arquivos com o mesmo nome nem duas pastas com o mesmo nome no Windows.

No final, temos a informação de quantos bytes são utilizados para armazenar os arquivos disponíveis nesta pasta (a conta é feita somente em relação aos arquivos) e, na última linha, o número de pastas do diretório atual e a quantidade de espaço disponível em disco. Agora, o cursor está pronto para que você digite um novo comando. Após a execução de um comando, o cursor fica disponível novamente para que o usuário continue seu trabalho.

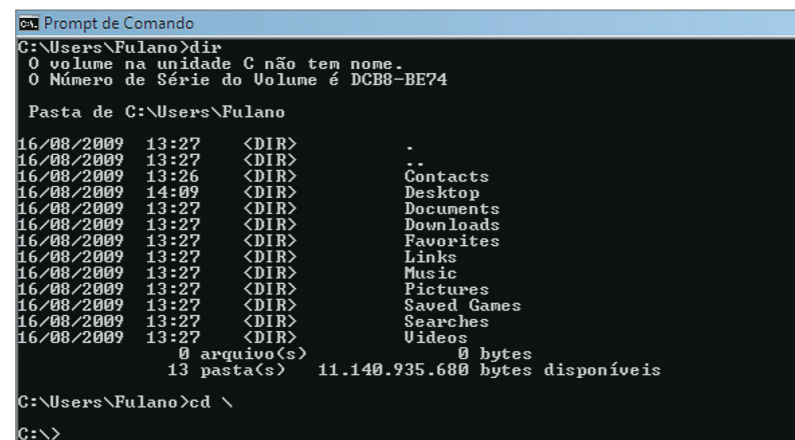
Renomear a unidade de disco é uma tarefa do administrador do sistema. Assim, é necessário executar o comando como administrador. Para fazer isso, antes de abrir o prompt, clique com o botão direito no atalho correspondente e escolha a opção “Executar como Administrador” (figura 181).

O comando “cd” (change directory, literalmente “mudança de diretório”) serve para navegarmos entre as pastas do sistema. No exemplo mostrado na figura 182, vamos navegar para o diretório raiz. Para isso digite o comando “cd \”.

Observe que o cursor mudou a informação sobre o diretório atual, ou seja, agora não estamos mais na pasta \Users\Fulano e sim na pasta raiz da unidade C.

Figura 182

Comando “cd \”.



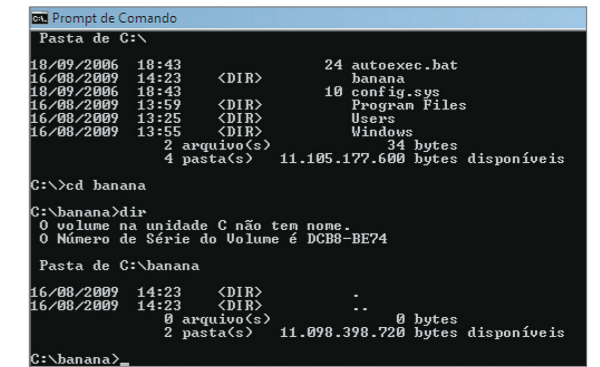
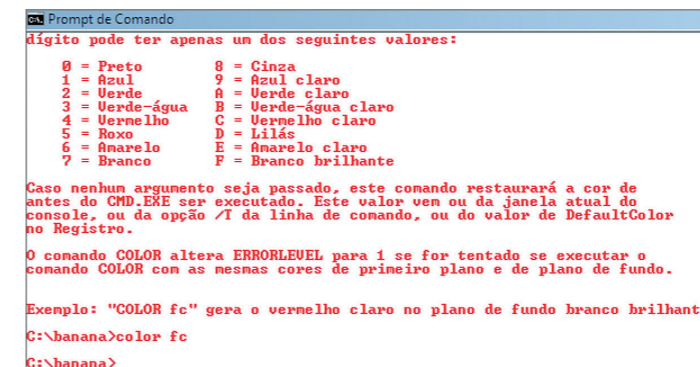
Comando	Em inglês	Tradução	Função
Color	Color	Cor	Muda a cor da letra e do fundo do prompt (para conhecer os códigos de cores, acesse a ajuda) (figuras 183 e 184).
Edit	Editor	Editor	Abre o editor de textos do prompt, que permite a criação de arquivos simples de textos dentro do próprio prompt (equivalente ao bloco de notas).
Copy	Copy	Copiar	Copia um arquivo para um diretório. Exemplo: copy teste.txt c:\users\fulano\pasta.
Move	Move	Mover	Move um arquivo para um diretório. Exemplo: move teste.txt c:\users\fulano\pasta.
Ren	Rename	Renomear	Renomeia o arquivo passado como parâmetro.
Date	Date	Data	Mostra a data atual e permite alterar a data do sistema.
Time	Time	Hora	Mostra a hora atual e permite alterar a hora do sistema.
Exit	Exit	Sair	Fecha o prompt de comandos.
Cls	Clear Screen	Limpar Tela	Limpa o prompt de comandos.

Tabela 3

Relação de comandos básicos.

7.3.2. Comandos básicos

Depois de aprender a utilizar o prompt de comandos é o momento de conhecer uma relação de seus comandos básicos (tabela 3). Todo comando possui uma página de ajuda na qual o usuário pode obter mais informações sobre como executá-lo e quais os recursos adicionais disponíveis. Sempre que quiser obter uma relação dos comandos básicos existentes no prompt, basta acionar o comando “help”. Veja exemplos do comando de cor (“color/?” e “color fc”) nas figuras 183 e 184.



7.3.3. Criando pastas pelo prompt

Depois de aprender a utilizar o prompt e de conhecer os caminhos para os diversos comandos, a tarefa de criar e excluir pastas pelo prompt se torna bastante fácil. Veja quais são os comandos de manipulação de pastas na tabela 4.

Figura 183

Resultado do comando “color/?”

Figura 184

Resultado do comando “color fc”.

Comando	Em inglês	Tradução	Função
md	Make directory	Criar diretório	Cria uma nova pasta ou diretório.
rd	Remove directory	Remover diretório	Exclui um diretório vazio.
dir	Directory	Diretório	Lista o conteúdo do diretório.
cd	Change directory	Mudar diretório	Navega entre os diretórios (precisa de um ou mais parâmetros, que podem ser o nome do diretório da pasta atual ou o caminho completo para o diretório).

Tabela 4

Comandos para diretórios, no Windows.

7.3.4. Comandos de configuração e gerenciamento

Há também comandos mais avançados, utilizados para alterar configurações do sistema (tabela 5).

Comando	Em inglês	Tradução	Função
ipconfig	Interface configuration	Configuração da interface (de rede)	Permite visualizar diversas informações da placa de rede, solicitar um novo ip de rede pelo parâmetro “/renew” e mostrar informações ainda mais detalhadas pelo parâmetro “/all”.
hostname	Host name	Nome da máquina	Retorna o nome do PC.
tree	Tree (figura 185)	Árvore	Mostra a árvore de diretórios disponível na unidade em que o comando foi executado.
shutdown	Shut down	Desligar	Desliga ou reinicia o PC a partir do parâmetro passado na linha de comando. Podemos configurar uma hora para que isso aconteça (ver shutdown !?).
title	Title	Título	Altera o título do prompt.
attrib	Attribute	Atributo	Altera os atributos de um arquivo. Por exemplo, podemos transformar um arquivo como somente leitura, executável, oculto etc.
systeminfo	System Information	Informações do Sistema	Retorna informações sobre o sistema operacional e o hardware.
tasklist	Task list	Lista de tarefas	Lista os processos em execução.
taskkill	Task kill	Mata tarefa	Usado para finalizar uma tarefa ou processo.

Tabela 5

Comandos de configuração e gerenciamento.

7.4. Prompt de comandos do Linux

Podemos trabalhar, também, com comandos no Linux. Para isso vamos utilizar um software chamado “Console” ou “Terminal”. Antes de tudo, é preciso entender que o Linux é um sistema operacional de modo texto. Ou seja, não passa de uma tela preta com a qual o usuário pode interagir por meio de comandos, como acontecia no antigo MS-DOS (figura 185). O que permite ao usuário clicar em janelas é um outro programa chamado “Interface Gráfica”, que no caso do Ubuntu é o Gnome.

A Interface Gráfica é executada automaticamente quando o Ubuntu é inicializado. Por isso, no nosso primeiro contato com o Linux, imaginamos que o sistema

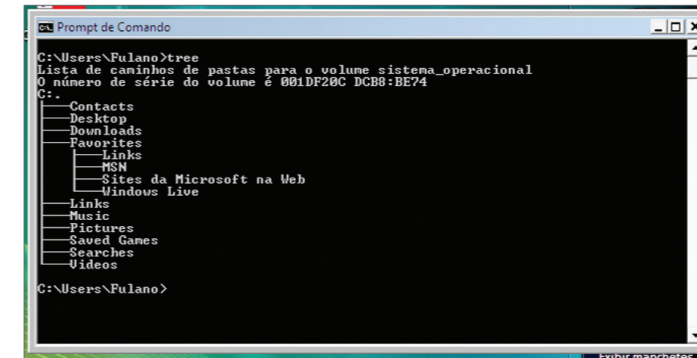


Figura 185
Comando “tree”.

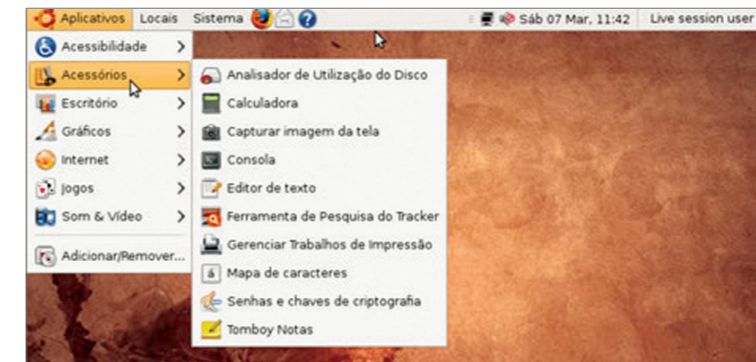


Figura 186
Como abrir o console gráfico.

operacional tenha aquela interface gráfica com menus, papel de parede e cursor do mouse, como o Windows, que já foi assim em antigas versões: eram de um programa de interface gráfica rodando sobre o MS-DOS.

Para começar a utilizar a linguagem de comandos no Ubuntu, clique no menu “Aplicativos / Acessórios / Console”, conforme indica a figura 186.

7.4.1. Terminal, Shell e tty

Uma janela como a da figura 187 aparecerá na tela. O nome correto do programa é “Console”, mas houve um erro de tradução para o português, corrigido nas novas versões do Ubuntu. Na 9.04, “Console” foi substituído por “Terminal”.

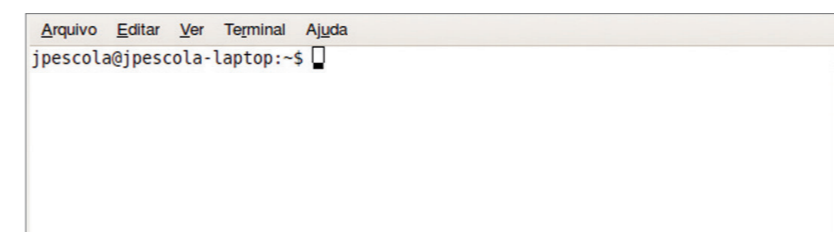
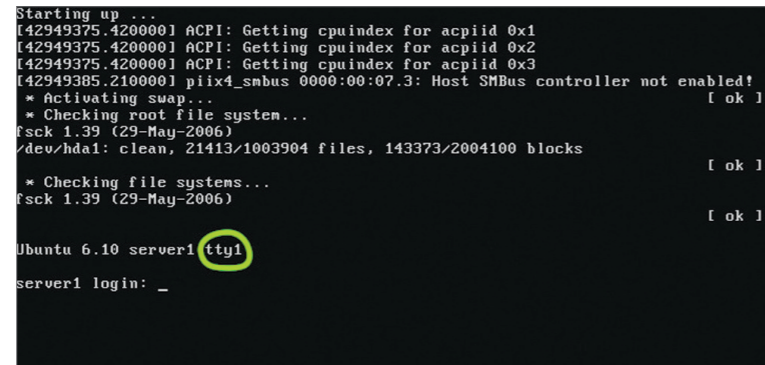


Figura 187
Terminal ou Console.

Figura 188

Imagem do Linux, sem interface gráfica.



DICAS

- Para executar um comando, basta digitá-lo e pressionar a tecla ENTER.
- Quando executamos um comando e nada retorna, significa que está tudo bem.
- Depois de executar um comando, o cursor aparecerá novamente, aguardando o próximo comando.
- Ao contrário do MS-DOS, o Linux é case sensitive. Isso quer dizer que ele diferencia letras maiúsculas e minúsculas em comandos, além de nomes de arquivos e/ou pastas.
- Para sair do console, execute o comando exit.

Quando é inicializado, como padrão, o Ubuntu disponibiliza sete terminais. Podemos ir para o primeiro (“tty1”) pressionando as teclas CTRL+ALT+F1.

Surgirá um terminal de modo texto, executado fora da interface gráfica. Essa seria a imagem típica do Linux, caso ainda não tivesse sido introduzida a interface gráfica.

O nome do terminal (mostrado na figura 188) é “tty1”. Pressionando CTRL+ALT+F2, iremos para o “tty2. Dessa forma, podemos utilizar até seis terminais diferentes e mais a interface gráfica que roda no sétimo “tty”. Para voltar à interface gráfica, basta pressionar CTRL+ALT+F7.

Antes de começar a trabalhar no console, é preciso saber alguns detalhes sobre ele. Se você está familiarizado com o MS-DOS ou com o prompt de comandos do Windows, provavelmente não terá dificuldades. Quando clicamos em “Aplicativos / Acessórios / Console”, abrimos um console de modo gráfico, programa que simula um verdadeiro console que poderia ser visualizado utilizando as teclas CTRL+ALT+F1 a F6. Esse instrumento pode facilitar a vida do usuário que não deseja sair da interface gráfica para digitar comandos.

Ao executar o TTY, visualizamos as informações exibidas na figura 189.

Figura 189

Informações do tty de modo texto.

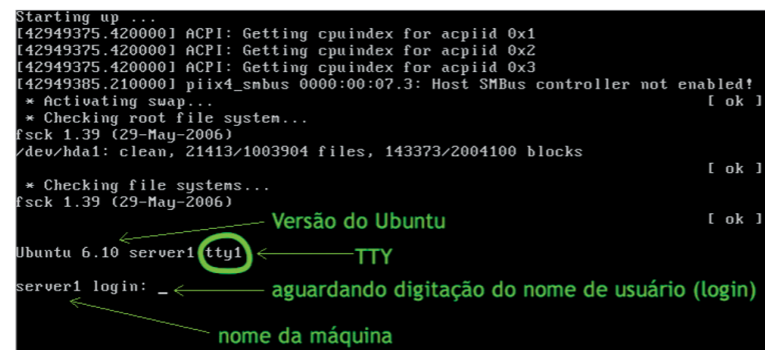
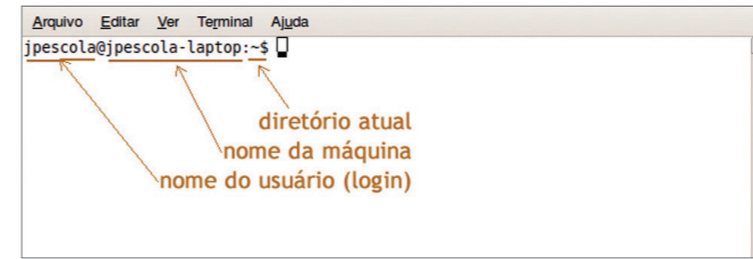


Figura 190

informações do terminal.



O TTY também solicita o nome de usuário e senha, tal como acontece com o Linux, quando se liga o computador. Primeiro vem o pedido do nome. Basta digitá-lo e pressionar “Enter” para que seja solicitada a senha (password). O console não exibirá os caracteres da password que estão sendo digitados, mas você pode continuar e pressionar “Enter”. Após digitar a senha (no caso de não estar utilizando o Live-CD ou o console gráfico), o console é apresentado na tela (figura 190).

No console, há um cursor piscando no final de um conjunto de informações que são exibidas o tempo todo quando trabalhamos com comandos. O valor que aparece antes do símbolo de arroba “@” é o nome do usuário que estamos utilizando (login). Depois do @ temos sempre outra informação: o nome da máquina, apresentado entre o @ e o sinal de dois-pontos “:”. Esse nome da máquina é configurado durante a instalação do sistema. Ou, no caso do Live-cd, denomina-se “Ubuntu”, assim como o usuário.

Entre os sinais de dois pontos “:” e cifrão “\$” temos o diretório atual, ou seja, a representação ou caminho para o diretório em que estamos no momento. O til “~” no Linux representa a pasta pessoal do usuário. Assim como na pasta “Meus Documentos” do Windows, a til do Linux serve para o usuário colocar seus arquivos pessoais. Na verdade, o til é somente uma representação do diretório /home/<usuário>. Para economizar caracteres e proporcionar outros benefícios (que veremos mais adiante), o console apresenta somente o caractere “~” em vez de /home/<usuário> entre os sinais de dois-pontos e cifrão.

7.4.2. Comandos básicos

Já aprendemos a entrar e sair de uma pasta. Se você utiliza ou já utilizou o prompt de comandos do Windows ou o MS-DOS, não deve encontrar dificuldades, até este ponto, mas há um detalhe importante a considerar.

Quando se usa o console, a primeira palavra digitada é considerada um COMANDO e todas as palavras escritas na mesma linha e separadas com espaço são PARÂMETROS (figura 191).

Um comando é separado de um parâmetro por um espaço, ou seja, toda vez que digitamos alguma coisa no console, sempre dando um espaço para, em seguida, digitar outra, a primeira palavra é um COMANDO e as demais, separadas por espaço, são os PARÂMETROS.

Quando utilizamos o Ubuntu por meio do Live-CD, não são solicitadas informações de usuário e senha. Nesse caso, o usuário padrão é o UBUNTU, que não possui senha. No console gráfico, o nome de usuário e senha não são solicitados, já foram informados quando você ligou o computador para entrar no sistema.

Figura 191

Comandos e parâmetros do terminal.

```
Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ cd Desktop/
jpescola@jpescola-laptop:~/Desktop$ ls
jpescola@jpescola-laptop:~/Desktop$ cd ..
jpescola@jpescola-laptop:~$ mkdir teste
jpescola@jpescola-laptop:~$ ls
Desktop Documentos Imagens Modelos Pública teste Vídeos
jpescola@jpescola-laptop:~$ mkdir pasta1 pasta2 pasta3
jpescola@jpescola-laptop:~$
```

Assim, quando digitamos o comando mkdir teste, MKDIR é um comando e TESTE, um parâmetro. Outro exemplo: se digitarmos “cd..”, a palavra CD é um comando e “..” é um parâmetro (figura 192).

Figura 192

Comandos e parâmetros.

```
Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ cd Desktop/
jpescola@jpescola-laptop:~/Desktop$ ls
jpescola@jpescola-laptop:~/Desktop$ cd ..
jpescola@jpescola-laptop:~$ mkdir teste
jpescola@jpescola-laptop:~$ ls
Desktop Documentos Imagens Modelos Pública teste Vídeos
jpescola@jpescola-laptop:~$ mkdir pasta1 pasta2 pasta3
jpescola@jpescola-laptop:~$ ls
Desktop Imagens pasta1 pasta3 teste
Documentos Modelos pasta2 Pública Vídeos
jpescola@jpescola-laptop:~$ rmdir pasta1
jpescola@jpescola-laptop:~$ ls
Desktop Documentos Imagens Modelos pasta2 pasta3 Pública teste Vídeos
jpescola@jpescola-laptop:~$
```

Nesse momento, talvez você tenha dúvidas em relação à criação de uma pasta com espaços como “meus documentos” ou “minhas músicas” no console. Veja o que acontece se digitarmos o comando “mkdir meus documentos” no console: ele cria uma pasta chamada “meus” e outra chamada “documentos”. É porque o console não entende que queríamos criar uma única pasta com o nome “meus documentos” e sim que a intenção era criar duas pastas diferentes (figura 193).

Uma dúvida que costuma aparecer neste momento diz respeito a como representar, no console, um nome de pasta com espaços (porque o espaço é justamente um caractere separador entre um comando e um parâmetro). A solução é simples. Para dizer ao console que um espaço deve ser entendido como um caractere do nome de um arquivo ou pasta, e não como um separador, basta digitar o caractere \ (barra invertida) antes do espaço (figura 194).

Figura 193

Criando a pasta “meus documentos”.

```
Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ mkdir meus documentos
jpescola@jpescola-laptop:~$ ls
Desktop Documentos meus pasta2 Pública Vídeos
documentos Imagens Modelos pasta3 teste
jpescola@jpescola-laptop:~$
```

Alguns comandos do Linux aceitam parâmetros, outros não. Para conhecer os parâmetros disponíveis nos comandos, basta digitar “man <comando>”, para abrir o manual de um **comando**. O manual tem todas as informações necessárias, inclusive os parâmetros aceitos e suas funções. O comando “is” mostra o conteúdo do diretório atual (figura 195).

Outro exemplo: o comando “cal” mostra o calendário do mês atual. No entanto, se passarmos como parâmetro o mês e o ano ou somente o ano desejado, obteremos um resultado diferente, conforme os parâmetros passados na linha de comandos (figura 196).

Para ler o manual de comando, utilize as teclas “Page Up” e “Page Down” ou as setas de navegação do teclado. Para sair e retornar ao console, pressione a tecla “Q”.

Figura 194

Criando a pasta meus documentos da maneira correta.

```
Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ mkdir meus documentos
jpescola@jpescola-laptop:~$ ls
Desktop Documentos meus pasta2 Pública Vídeos
documentos Imagens Modelos pasta3 teste
jpescola@jpescola-laptop:~$ mkdir meus\ documentos
jpescola@jpescola-laptop:~$ ls
Desktop Documentos meus Modelos pasta3 teste
documentos Imagens meus documentos pasta2 Pública Vídeos
jpescola@jpescola-laptop:~$
```

Figura 195

Manual do comando “is”.

```
Arquivo Editar Ver Terminal Ajuda
LS(1)
LS(1)
NOME
ls, dir, vdir - lista o conteúdo do diretório
SINOPSE
ls [opções] [arquivo...]
dir [arquivo...]
vdir [arquivo...]
Opções POSIX: [-CFRaCdilqrтуl]
Opções GNU (forma reduzida): [-labcdfghiklmnopqrstuvwXABCDFGHLNORSUX]
[-w coluna] [-T coluna] [-I modelo] [--full-time] [--show-control-chars]
[--block-size=size] [--format={long,verbose,com-mas,across,vertical,single-column}] [--sort={none,time,size,extension}]
[--time={atime,access,use,ctime,status}] [--color={none,auto,always}]
[--help] [--version] [--]
DESCRIÇÃO
O programa ls lista primeiramente seus argumentos que não sejam arquivos de diretórios, e, então, para um argumento diretório, todos os arquivos listáveis incluídos dentro daquele diretório. Se nenhum argu-
Manual page ls(1) line 1
```

Figura 196

O comando “cal”.

```
Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ cal
julho 2009
Do Se Te Qu Qu Se Sá
          1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

jpescola@jpescola-laptop:~$ cal jan 2010
janeiro 2010
Do Se Te Qu Qu Se Sá
          1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
31
jpescola@jpescola-laptop:~$
```

Figura 197

Comando "gedit" para abrir o editor de textos.

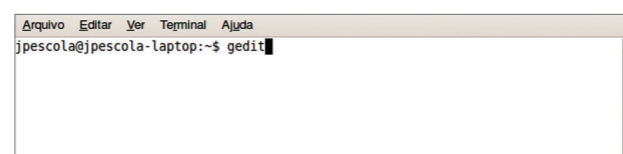
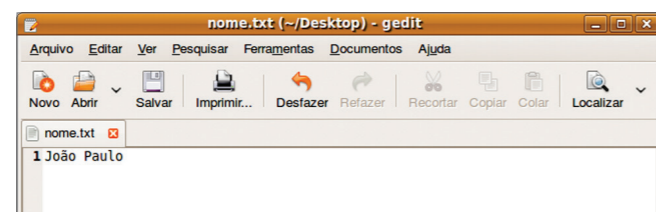


Figura 198

Digitando seu nome no "Gedit".



7.4.3. Criação de arquivos texto no Linux, sem usar editores

O próximo passo é criar arquivos do tipo texto. Eles são bastante utilizados no Linux ou em qualquer outro sistema operacional para as mais diversas necessidades. No Linux, particularmente, os arquivos texto servem para configurar softwares. Ao contrário do Windows, as configurações dos softwares instalados no Linux são feitas por meio de arquivos simples do tipo texto (padrão ASCII), como os que criamos no Bloco de Notas do Windows. No caso do Linux, é possível utilizar um editor de textos simples, como o "Gedit", que só pode ser executado no console gráfico e é o padrão do Gnome. Basta, então, digitar "gedit" no console gráfico (figura 197).

O Gedit também pode ser executado por meio do menu Aplicativos / Acessórios / Editor de Textos. Digite seu nome no "gedit" (figura 198).

Salve o arquivo na pasta Desktop com o nome "nome.txt" (figura 199). Feche o Gedit, minimize todas as janelas abertas e você verá o seu novo arquivo gravado na área de trabalho do Ubuntu (figura 200).

Figura 199

Salvando o arquivo do "Gedit".

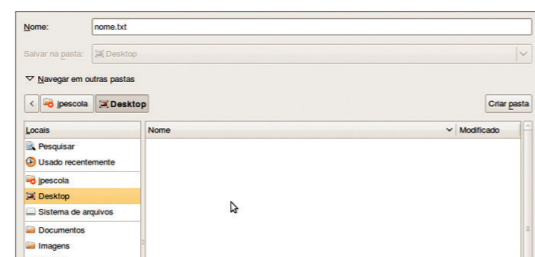


Figura 200

Arquivo salvo na área de trabalho.

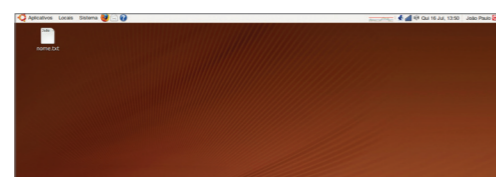


Figura 201

Visualizando o arquivo salvo na pasta "Desktop" (área de trabalho).

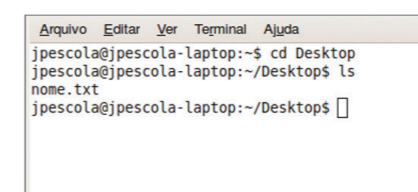
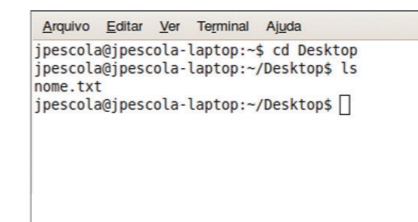


Figura 202

Visualizando o conteúdo de um arquivo-texto.



Para visualizar o conteúdo do arquivo-texto no console, abra-o e digite o comando "cd Desktop" para entrar na pasta Desktop (área de trabalho).

Perceba que a pasta "Desktop" está dentro de sua pasta pessoal (figura 201). Por isso, você pode salvar seu novo arquivo nesse local. Digite o comando "ls" para listar o conteúdo da pasta "Desktop".

Observe que seu arquivo realmente está na pasta. Para visualizar o conteúdo, há duas opções:

- a) Utilizar o Gedit, digitando o comando "gedit nome.txt" (faça o teste).
- b) Utilizando o comando "cat", que mostra o conteúdo de um arquivo-texto: "cat nome.txt".

O comando "cat" mostra o conteúdo de um arquivo-texto (figura 202). Para criar um novo arquivo do tipo texto no console, utilizamos o comando mostrado na figura 203.

Use o comando "cat" para visualizar o conteúdo de seu arquivo recém-criado (figura 204).

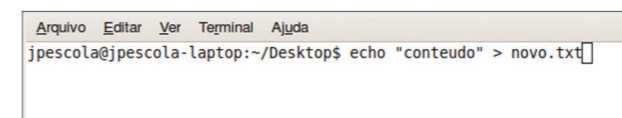


Figura 203

Criando um arquivo do tipo texto no console.

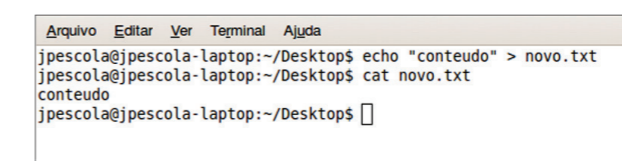


Figura 204

Visualizando o conteúdo do arquivo recém-criado "novo.txt".

Figura 205

Adicionando uma nova linha ao arquivo-texto.

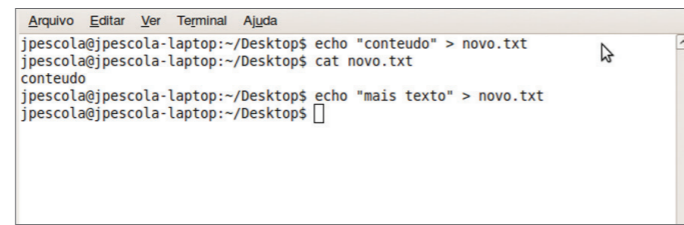
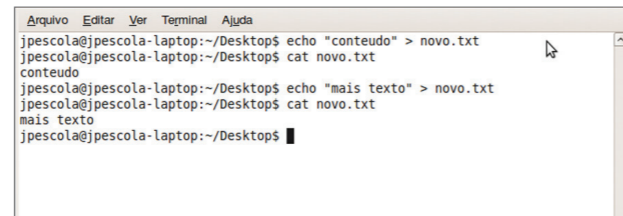


Figura 206

Visualização do conteúdo do arquivo, após adicionar uma nova linha.



Agora vamos adicionar mais uma linha no arquivo “novo.txt” (figura 205). E visualizar o arquivo (figura 206).

Você notará que o conteúdo anterior do arquivo sumiu. Isso acontece porque se usa somente o sinal de maior “>” para direcionar a saída do comando para o arquivo. Veja como funciona:

- Utilizando um sinal de maior (>), o Console cria o arquivo (caso ele não exista). Ou então substitui seu conteúdo existente.
- Utilizando dois sinais de maior (>>), o Console cria o arquivo (se ainda não existir). Ou incrementa o conteúdo existente.

Vamos fazer o teste utilizando dois sinais de maior (figura 207).

Você pode até achar inútil criar arquivos-texto no console, mas saiba que esse tipo de procedimento é muito comum em scripts shell (arquivos de lote do Linux) para criar arquivos de log (registro) no sistema operacional.

Podemos também direcionar uma saída de um comando para um arquivo-texto na forma apontada na figura 208.

Figura 207

Dois sinais de maior (>>) para acrescentar conteúdo em arquivo já existente.

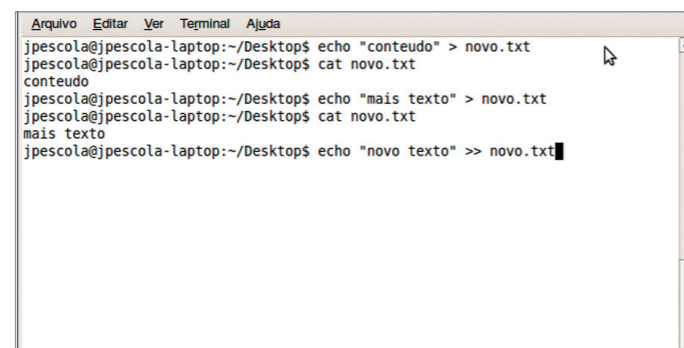


Figura 208

Comando “date”.

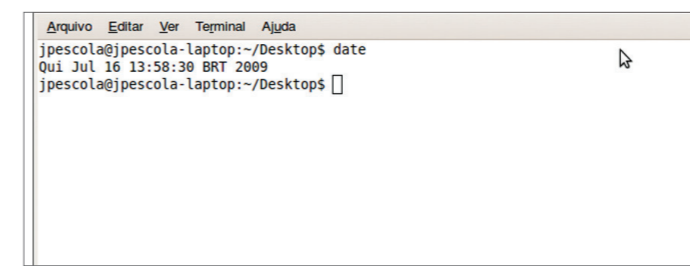
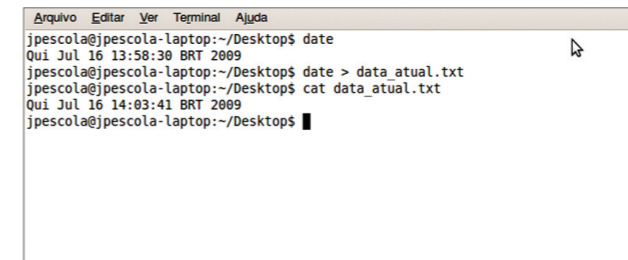


Figura 209

Adicionando o retorno de um comando em um arquivo-texto.



O comando “date” retorna a data e a hora atual na tela. Porém, também é possível fazer com que o console retorne a data e a hora atuais em um arquivo, e não na tela.

Vamos agora direcionar a saída do comando “date” (figura 209).

Agora o console criou um arquivo com o nome data_atual.txt, cujo conteúdo é justamente a saída do comando “date”, ou seja, a data atual.

7.4.4. Comandos de configuração e gerenciamento

Alguns comandos são especialmente úteis na administração de um computador, pois ajudam a descobrir o que está acontecendo no sistema. O primeiro que vamos estudar é o que mostra o total de memória livre e em uso: o comando “free” (figura 210).

Normalmente ele apresenta a memória disponível em Kilobytes, mas é possível alterar a unidade de saída utilizando o parâmetro “-m” (Megabytes) ou “-g” (Gigabytes).

Figura 210

Comando “free”.

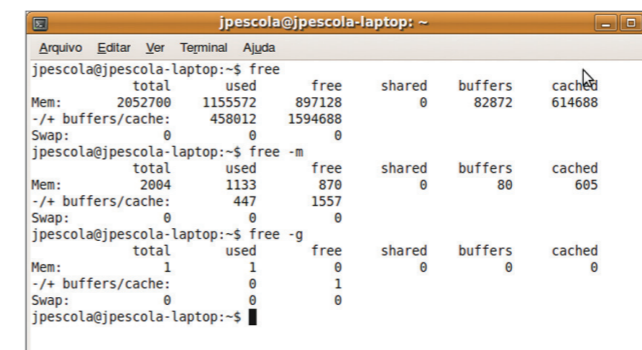


Figura 211
Comando “df”
(disk free).

```

Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ df
Sist. Arq.          1K-blocos   Usad  Dispon.   Uso% Montado em
/dev/sda1          29293564 14955196 14338368  52% /
tmpfs              1026348    0      1026348    0% /lib/init/rw
varrun            1026348    540     1025808    1% /var/run
varlock           1026348    0      1026348    0% /var/lock
udev              1026348    140     1026208    1% /dev
tmpfs             1026348    232     1026116    1% /dev/shm
lrm               1026348    2192    1024156    1% /lib/modules/2.6.28-13-g
eneric/volatile
/dev/sda2         126989912 74915788 52074124  59% /dados
jpescola@jpescola-laptop:~$ df -h
Sist. Arq.          Tam  Usad  Disp  Uso% Montado em
/dev/sda1          28G   15G   14G   52% /
tmpfs              1003M    0 1003M    0% /lib/init/rw
varrun            1003M  540K 1002M    1% /var/run
varlock           1003M    0 1003M    0% /var/lock
udev              1003M  140K 1003M    1% /dev
tmpfs             1003M  232K 1003M    1% /dev/shm
lrm               1003M  2,2M 1001M    1% /lib/modules/2.6.28-13-generic/volat
ile
/dev/sda2         122G   72G   50G   59% /dados
jpescola@jpescola-laptop:~$
    
```

DICA
Não é possível gravar arquivos em uma partição do tipo SWAP, que só pode ser utilizada pelo sistema operacional para trocas de dados com a memória RAM.

Também é possível visualizar a quantidade de espaço livre em disco. Para isso utilizamos o comando “df” (disk free) (figura 211). Da mesma forma que fizemos com o comando free, podemos converter as unidades de saída do comando “df”. O parâmetro “-h” (human) retorna as unidades de forma mais amigável. Pode ser em Kilobytes, Megabytes ou Gigabytes.

O comando “du” (figura 212) retorna o tamanho da pasta no disco rígido, mostrando a quantidade de Kilobytes necessária para armazenar determinada pasta no disco. Pode ser utilizado sem parâmetros (retornará a quantidade de Kilobytes utilizados por todas as pastas do diretório atual) ou tomando como parâmetro a pasta desejada.

O comando “uptime” (figura 213) mostra diversas informações importantes: hora atual, há quanto tempo o PC está ligado, número de usuários logados na máquina etc.

O comando “top” mostra um monitor dos processos que estão em execução no computador, entre outras informações (figura 214). Para sair do top, basta pressionar a tecla “Q” do teclado.

Figura 212
Comando “du”
(disk usage).

```

Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ du Desktop
0 Desktop/.fr-L4ZvJM
12 Desktop
jpescola@jpescola-laptop:~$
    
```

Figura 213
Comando
“uptime”.

```

Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ uptime
16:54:11 up 6:24, 2 users, load average: 0.30, 0.16, 0.10
jpescola@jpescola-laptop:~$
    
```

Figura 214:
Comando “top”.

```

Arquivo Editar Ver Terminal Ajuda
top - 16:54:43 up 6:25, 2 users, load average: 0.61, 0.24, 0.13
Tasks: 147 total, 1 running, 146 sleeping, 0 stopped, 0 zombie
Cpu(s): 3.2%us, 1.6%sy, 1.0%ni, 93.7%id, 0.0%wa, 0.2%hi, 0.3%si, 0.0%st
Mem: 2052700k total, 1680408k used, 372292k free, 83452k buffers
Swap: 0k total, 0k used, 0k free, 1130472k cached

  PID USER      PR  NI  VIRT  RES  SHR  S %CPU  MEM%   TIME+  COMMAND
 3374 root        20   0 362m  50m  17m  S   4.2   2.5  13:39.57 Xorg
 5047 jpescola   20   0 1381m 234m  15m  S   4.1  11.7  13:07.23 java
 3966 jpescola   25   5 30768 8464 5884  S   3.0   0.4   6:20.98 hardware-monito
 5369 jpescola   20   0 34220  10m 5908  S   0.0   0.5   0:04.64 gnome-terminal
    
```

7.5. Sistemas de arquivos

Um sistema operacional precisa ter um sistema de arquivos. Só assim poderá gerenciar, por exemplo, a gravação de um novo dado no disco rígido, garantindo que ela esteja disponível para acesso, independentemente do local de armazenado no HD. Graças a esse recurso, o sistema operacional consegue informar ao usuário qual o espaço disponível em disco. Também permite ao HD gravar um mesmo arquivo em partes espalhadas pelo disco e recuperá-las quando necessário.

7.5.1. Sistemas de arquivos do Windows (NTFS e Fat32)

Cada sistema operacional tem um sistema de arquivos particular. No caso do Windows, é o NTFS, mas o padrão já foi o Fat32. Os primeiros sistemas de arquivos utilizados pelo Windows foram o Fat (File Allocation Table, ou Tabela de Alocação de Arquivos). Posteriormente foi lançado o NTFS (New Technology File System, ou nova tecnologia em sistema de arquivos), que traz diversas características como segurança, capacidade de armazenamento e limitações aprimoradas em relação às versões anteriores. Se você for formatar um pen-drive ou um disco rígido pequeno, poderá escolher um dos sistemas de arquivos já mencionados. Entretanto, para discos de grande capacidade – e por se tratar de um instrumento mais novo e confiável –, recomenda-se a utilização do NTFS.

7.5.2. Sistemas de arquivos no Linux (Ext3 e ReiserFS)

No caso do Linux, existem diversos sistemas de arquivos disponíveis. A maioria das distribuições, porém, adota o Ext3 (third extended file system, ou terceiro sistema estendido de arquivos). É uma evolução do Ext2 (second extended file system, ou segundo sistema estendido de arquivos), que era o mais comum nas distribuições. O Ext3 traz o recurso de Journaling (registro de dados como em um jornal), no qual as informações são salvas automaticamente durante a execução do sistema operacional. Elas podem ser recuperadas em caso de falha ou queda de energia elétrica. Outro sistema de arquivos bastante usado no Linux é o **ReiserFS** (Reiser File System), que também oferece Journaling, mas somente retém as informações de cabeçalhos – não faz o registro de dados, o que o torna mais rápido que o Ext3.

7.5.3. Sistema de arquivos para memória virtual no Linux (Swap)

Quando um computador fica sem memória RAM, acaba recorrendo a um recurso chamado “memória virtual”. Ela permite copiar o espaço de memória mais antigo para um arquivo no disco rígido, provocando um “desafogamento” da memória RAM. Assim, novos programas podem ser abertos, mesmo quando a memória da máquina está escassa.

O ReiserFS foi criado pelo norte-americano Hans Thomas Reiser, um programador de computadores famoso no mundo do software livre. Ele é fundador da Namesys, empresa de desenvolvimento de software especializada em sistemas operacionais.

Durante o processo de instalação no Linux, uma pequena partição é criada em disco (normalmente do tamanho da memória RAM instalada), cujo sistema de arquivos é o chamado “SWAP” (troca).

DICA

Ao montar uma partição, lembre-se de criar a pasta de destino, caso não haja uma. Sem isso, o comando não vai funcionar.

Uma partição “SWAP” entra em ação toda vez que a memória RAM fica escassa. Funciona como a memória virtual no Windows. Porém, nesse caso não temos um arquivo de memória e sim uma partição inteira em disco disponível. Este é um dos motivos pelos quais o desempenho de um computador Linux geralmente é melhor do que o mesmo equipamento que utiliza o Windows.

7.6. Montagem de dispositivos (partições e disquetes) no Linux

Cada sistema operacional tem características únicas, já que foram desenvolvidos por pessoas e empresas diferentes. A afirmação vale, inclusive, para as evoluções constantes desses sistemas. Por exemplo: há alguns anos, acessar os dados de um pen-drive, CD ou disquete no Linux era tarefa complexa. Os usuários precisavam montar discos por meio do comando “mount” para poder utilizá-los no sistema operacional – muito diferente do Windows, em que basta colocar o CD ou disquete no drive e clicar no ícone ou atalho correspondente para acessá-lo. Até por isso, muitos usuários iniciantes desistem do Linux. A boa notícia é que a maioria das distribuições Linux já corrigiu esse problema, permitindo que cada disquete ou CD seja acessado da mesma forma que em outros sistemas operacionais pelo usuário ao inseri-lo no drive.

Entender como isso funciona requer um estudo do processo de montagem de dispositivos no Linux, que trabalha com uma forma de acesso a discos diferente da do Windows. Para acessar os dados de um drive externo é preciso montar o dispositivo, ou seja, mapeá-lo em uma pasta já existente no sistema de arquivos.

7.6.1. Uso de dispositivos de armazenamento no Linux

Durante a inicialização, o sistema operacional Linux monta, automaticamente, os dispositivos ou partições de disco nas pastas (diretórios) correspondentes. Nas novas distribuições (como Ubuntu), pen-drives e CDs, esses dispositivos são montados quando inseridos no drive ou porta USB. Esse processo se dá também pelo próprio sistema operacional.

Existe um arquivo no qual o usuário pode configurar quais partições ou discos devem ser montados durante a inicialização do sistema: /etc/fstab (file system table),

Figura 215

Conteúdo do arquivo “fstab” armazenado na pasta “/etc”.

```

Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'vol_id --uuid' 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>
proc /proc proc defaults 0 0
# / was on /dev/sda1 during installation
UUID=5b183e4b-c262-4889-add3-d1178d04fab1 / reiserfs notail,relatime 0 1
# /dados was on /dev/sda2 during installation
UUID=c8a1521e-8178-46ea-b057-036654d50f85 /dados reiserfs relatime 0 2
/dev/scd0 /media/cdrom0 udf,iso9660 user,noauto,exec,utf8 0 0
jpescola@jpescola-laptop:~$
    
```

como ilustra a figura 215. Para saber mais, acesse o manual do “fstab”, digitando no console: “man fstab”.

7.6.2. Montagem de dispositivos (partições e disquetes)

Quando usamos o console ou o terminal, podemos montar e desmontar dispositivos manualmente por meio dos comandos “mount” e “umount” (figura 216).

Não é possível montar nenhum dispositivo no prompt de comandos sem autorização. É que o comando mount só pode ser executado pelo administrador (root). Para executar um comando como administrador, é preciso adicionar o comando “sudo” (abreviação para “super user do”, literalmente “fazer como superusuário”) à frente do comando restrito.

No comando da figura 216, montamos a partição “sda1” na pasta “hd”, uma subpasta de “/media”. Ao listar o conteúdo da pasta, você terá o da partição “sda1”, que representa a primeira partição do primeiro disco instalado na máquina (tabela 6).

```

Arquivo Editar Ver Terminal Ajuda
jpescola@jpescola-laptop:~$ cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'vol_id --uuid' 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>
proc /proc proc defaults 0 0
# / was on /dev/sda1 during installation
UUID=5b183e4b-c262-4889-add3-d1178d04fab1 / reiserfs notail,relatime 0 1
# /dados was on /dev/sda2 during installation
UUID=c8a1521e-8178-46ea-b057-036654d50f85 /dados reiserfs relatime 0 2
/dev/scd0 /media/cdrom0 udf,iso9660 user,noauto,exec,utf8 0 0
jpescola@jpescola-laptop:~$ mount /dev/sda
sda sda1 sda2
jpescola@jpescola-laptop:~$ mount /dev/sda1 /media/hd
mount: operação exclusiva de root
jpescola@jpescola-laptop:~$ sudo mount /dev/sda1 /media/hd
    
```

O “sudo” permite executar qualquer comando como administrador. Está disponível a quem instalou a distribuição na máquina. Na primeira vez que o “sudo” for executado, o terminal vai pedir a senha para verificar se o usuário realmente tem permissão.

Figura 216

Montando uma partição do disco em uma subpasta de /media.

O comando “mount” precisa de pelo menos dois parâmetros: o dispositivo que será montado e a pasta de destino. Depois de usar o dispositivo, basta sair da pasta de montagem e partir para o comando “umount”, a fim de desmontá-lo. Como parâmetro de caminho, informe o dispositivo (“/dev/sda”) ou a pasta em que ele foi montado.

Dispositivo	Descrição
/dev/sda1	Primeira partição do primeiro disco
/dev/sda2	Segunda partição do primeiro disco
/dev/sda3	Terceira partição do primeiro disco
/dev/sdb1	Primeira partição do segundo disco (pode ser seu pen-drive)
/dev/sdc1	Primeira partição do terceiro disco
/dev/cdrom	Drive de CD-ROM
/dev/cdrw	Drive de CD-RW ou DVD-RW
/dev/fd0	Primeiro drive de disquete
/dev/fd1	Segundo drive de disquete

Tabela 6

Comandos de montagem de dispositivo.