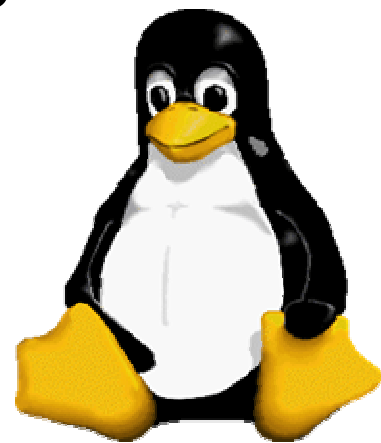


**Curso Básico**

# Linux

---

**Módulo Único**



# Conectiva Linux 4.0

## Versão Guarani



### ÍNDICE:

01.	Histórico	02
02.	Linux como Sistema Operacional	04
03.	Onde adquirir o Linux ?	06
04.	Hardware necessário	06
05.	Entrando e saindo do Sistema	06
06.	Ambiente de Trabalho (Shell)	08
07.	Contas e senhas	09
08.	Consoles Virtuais	11
09.	Comandos Básicos	11
10.	Editando no Linux (modo texto)	13
11.	Usando compactadores	13
12.	Controle de Acesso de Arquivos	16
13.	CD-ROM's, Disquetes, Discos Rígidos e Sistema de Arquivos – Visão Geral	18
14.	Usando Linux e Windows no mesmo disco rígido	20
15.	Interfaces Gráficas	20
16.	Informações Úteis	22



## 1 - INTRODUÇÃO

Nos anos 50, quem dissesse que Rockefeller um dia perderia toda a sua fortuna seria tido como um louco. Um cidadão da década seguinte não conseguiria imaginar nada mais assombroso que a TV em preto e branco. Imagine então com o computador!!!

Os anos 80's e 90's viram surgir um dos maiores impérios empresariais de todos os tempos, dirigido por um pacato sujeito chamado William Henry Gates III, que veio a tornar-se o homem mais rico do mundo.

Sr. Gates conseguiu o feito graças a uma de suas invenções, um programa de computador que facilitava a operação desses intrincados aparelhos. As inegáveis qualidades da criação de Bill Gates foram notadas no mundo todo, e seu programa, chamado Windows, dominou 90% das máquinas em atividade.

Desde simples usuários, passando por programadores e analistas, chegando aos gerentes, todos só enxergavam uma solução para todos os problemas propostos pela computação: aquela que vem daquela empresa (diga-se Microsoft).

Haviam naquela época (e ainda há!!!) outras opções de sistemas (Unix, Xenix, BSD, por exemplo), mas eram de custos inacessíveis, além de não ser aquilo que todo mundo sonhava. Todos procuravam alguma alternativa que fosse confiável, barata, adequada à situação, à prova de falhas e extensível para futuras mudanças de realidade. Até que um dia...



Linus Torvalds

Mais propriamente no mês de Agosto de 1991, um pacato e jovem geek de 21 anos de idade, iniciou o projeto "LINUX". Seu nome: Linus Torvalds, então estudante de Ciência da Computação da Universidade de Helsinque, capital da Finlândia, um mero "nerd".

Usando o Assembler (ou Assembly, como queira...), Linus iniciou cortando(hacking) o kernel como um projeto particular, inspirado em sua paixão pelo Minix, um pequeno sistema UNIX, desenvolvido por Andy Tannenbaum.

O estudante universitário desejava desenvolver uma versão do Unix que rodasse em micros PC AT e compatíveis, mas que fosse diferente dos sistemas Unix já existentes, cujo preço era exorbitante para o usuário comum.

Linus chegou a divulgar a idéia num newsgroup de que participava (sem êxito...) e embalado pelo projeto, programou sozinho a primeira versão do kernel do Linux(núcleo do sistema operacional).

Ele se limitou a criar, em suas próprias palavras, "um Minix melhor que o Minix" ("a better Minix than Minix").

Até que numa calma manhã do dia 05 de Outubro de 1991, Linus anunciou a primeira versão "oficial" do Linux, versão 0.02 .

Depois de finalizar o kernel, Linus deu ao seu filhote o rumo que desencadeou seu grande sucesso: passou a distribuir o código-fonte do kernel pela internet (comp.os.minix) para que outros programadores pudessem aprimorar o sistema.

Ele colocava a seguinte mensagem:

**"Você suspira por melhores dias do Minix 1.1, quando homens serão homens e escreverão seus próprios "device drivers" ? Você está sem um bom projeto e está morrendo por colocar as mãos em um S.O. no qual você possa modificar de acordo com suas necessidades? Você está achando frustrante quando tudo trabalha em Minix ? Chega de atravessar noites para obter programas que trabalhem corretos? Então esta mensagem pode ser exatamente para você ?**

**Como eu mencionei a um mês atrás, estou trabalhando em uma versão independente de um S.O. similar ao Minix para computadores AT-386. Ele está, finalmente, próximo ao estágio em que poderá ser utilizado(embora possa não ser o que você esteja esperando), e eu estou disposto a colocar os fontes para ampla distribuição. Ele está na versão 0.02..., contudo, eu tive sucesso rodando bash, gcc, gnu-make, gnu-sed, compressão, etc. nele."**

Assim, várias empresas e programadores de todo o planeta contribuíram com seus conhecimentos para melhorar o Linux.

Mais que um sistema operacional, o Linux é a representação prática de uma nova filosofia de distribuição e produção de software. O Linux é "gratuito"(é um prazer dizer isso!!!). E mais: seu código fonte ainda está completamente aberto, para que programadores de todo planeta possam modificá-lo. Há dez anos, ninguém ousaria prever algo desse tipo. Como pode alguém gastar horas e mais horas em um trabalho para depois largá-lo na internet, sem controle, sujeito às mais diferentes alterações?

O que alguns julgavam a ruína, provou ser o grande trunfo do Linux.



A lógica é simples: distribua um produto de graça, deixe que os consumidores façam nele as alterações que quiserem, e assim você terá algo pulsante, em constante e inexorável evolução. Esse foi o segredo: trabalho cooperativo e voluntário. Linus distribuiu seu trabalho sem cobrar nada e em troca, exigiu que os outros programadores envolvidos no projeto fizessem o mesmo. Por isso é gratuito. A união fez a força - fez o LINUX.

O Linux, atualmente, tem recebido apoio de várias empresas como Netscape, Corel, Sun, Borland(dona do Delphi), Intel e Oracle. Todas usam Linux e desenvolvem produtos para Linux. As estimativas de seu uso variam entre 10 e 15 milhões de computadores. Ele tem ganho aceitação e propaganda no mundo inteiro. Em 1997, 105 computadores Alpha Digital com Linux, ligas em rede, renderizaram as cenas do filme "Titanic", durante 3 meses, ininterruptamente. Em quase todas as grandes empresas do mundo, há pelo menos um sistema Linux instalado.

- **Quem utiliza Linux?**

Ao redor do planeta se estima que tenhamos mais de vinte milhões de usuários Linux. Sendo que no Brasil, o País que apresentou o maior índice de crescimento no primeiro semestre de 99, este número gira ao redor de 400 mil usuários.

Dentre os mais diversos usuários, podemos ressaltar alguns mais conhecidos: NASA, Exército Americano, Governo da Itália, Governo da Califórnia, fábricas de robôs na Suécia, hospitais na França, praticamente todas as Universidades, Ministério da Saúde, Correio Norte Americano, etc...

No Brasil é bastante difundido no meio acadêmico, em empresas de desenvolvimento de



O Linux de Linus Torvalds, não contém nenhuma linha de código do Unix. Mas o Linux foi escrito para ser de acordo com o padrão API POSIX, o mesmo do Unix (uma espécie de ISO ou ABNT). Por isso se diz que o Linux é um Unix.

Por causa da API POSIX, do conjunto de utilitários e do uso do X-Window, o Linux é tão parecido com o Unix que existem empresas que usam o Linux para desenvolver para Unix que não seja o dela mesma (por exemplo, a IBM). Veja que a Microsoft está tentando transformar o NiceTry em um Unix. Ela espera que algum dia no futuro seja um Unix melhor que os outros, algo assim tipo o Linux. E para isto está aproximando-o do padrão API POSIX.

O Linux possui todas as características que você pode esperar de um Unix moderno, incluindo:

- Multitarefa Real;
- Memória Virtual;
- Biblioteca compartilhada;
- "Demand loading";
- Gerenciamento próprio de memória;
- Executáveis "copy-on-write" compartilhados;
- Rede TCP/IP (incluindo SLIP/PPP/ISDN);
- X Window; etc.



Posso até fazer uma bela analogia com o Windows: Sabe aquela mina linda que passa por você todas manhãs junto com uma outra que você nem presta atenção? Depois de conhecer as duas você acaba se apaixonando pela outra por ela ser confiável (fiel), robusta (saudável :) e inteligente, enquanto a bonita é flácida, burra, e muda de opinião a cada cinco minutos, sem contar que ela tem um tique esquisito, que os médicos dizem que é uma tal de síndrome de GPF).

## • - Faq's

### a - Como se pronuncia LINUX ?

Por incrível que pareça, não é "lainucs". Linux se pronuncia "linucs" exatamente como se lê. O usuário Linux chama-se "linuxer".

### b - Como obter suporte ?

Existem várias maneiras. Você pode entrar numa lista de discussão, participar de algum fórum relacionado ao Linux, ou até mesmo num chat. A maioria é na Web mesmo... Até o final da edição desta apostila não havia nenhuma empresa brasileira que vendesse suporte a Linux.

### c - Como usuário do Linux, tenho algum direito sobre ele ?

Claro...leia abaixo:

\* TODOS TEM O DIREITO DE PERGUNTAR QUALQUER COISA - E o dever de antes disso ter pesquisado sobre o assunto e não ter obtido uma resposta. E pergunte no lugar certo. Não pergunte uma coisa básica numa sessão avançada, nem pergunte sobre software numa sessão de hardware.

\* TODOS TEM O DEVER DE AJUDAR O PRÓXIMO - Bom...se chegamos até aqui é por que ensinaram a gente. Faça o mesmo! Assim o clã dos linuxers cresce e seu nome também. Se todos gostarem do Linux, vamos ter todos os programas convertidos para o Linux, por exemplo.

\* TODOS TEM O DIREITO DE COPIAR - Podemos copiar tudo, respeitando os nomes dos autores e as licenças dos programas. Procure em algum faq sobre a licença do Linux (se acha que eu ia dar de lambuja ?!?! Vá se acostumando....)

### **d - Como devo me vestir? O que devo comer? O que devo comprar?**

Faça tudo aquilo que você quiser. O Linux é isso, liberdade!! Não existe um tipo específico de gente que usa Linux. É só usar e pronto! Você quer liberdade maior do que não comprar um programa, alterá-lo de tudo quanto é jeito, emprestar o CD pra todo mundo, não ser preso e descobrir que este é o melhor programa que você conhece? Venha para onde está o sabor...

### **3 - ONDE ADQUIRIR O LINUX ?**

Uma opção é fazer o download na internet. Você pode apontar o seu browser para:

<http://www.sunsite.unc.edu/pub/LINUX/distributions>

e escolha sua versão do Linux (RedHat, Slackware, etc.)

Mas, se você preferir não gastar suas horas na internet (uma noite inteira e mais um pouco), você pode optar pelo pacote distribuído pela Conectiva ([www.conectiva.com.br](http://www.conectiva.com.br)). O valor cobrado de R\$ 88,00 se refere a 3 CD's contendo o Linux em português e com mais de 800 aplicativos + manual de instalação em português + Guia do Usuário com mais de 750 páginas de documentação Linux em português. Quem preferir apenas os CD's, o preço é de R\$ 28,00. Lógico que para quem estiver iniciando, o pacote completo seria o mais adequado. Dentre os aplicativos, encontra-se a suíte Star Office 5.1 traduzida para o português, e cujos programas lêem arquivos do Microsoft Office (...que beleza !!!). O site da Conectiva é:

<http://www.conectiva.com.br>

Você deve estar se perguntando: "Mas o Linux não é de graça ?" Sim...o Linux é de graça. O que a Conectiva cobra se refere apenas ao trabalho de traduzir o Linux, passar mais de 800 programas para os CD's, editar e imprimir um manual de quase 800 paginas em português pra gente. Só isso ...!!! Não acho que você iria gastar uma noite inteira na Web (+ conta telefônica) pra pegar uma versão em inglês, não é ?

Atualmente, a Conectiva está distribuindo a versão 4.0 do Linux Guarani(não confunda com a versão do kernel: 2.2.5), que é a versão que usaremos no curso (yesssssssss !!!!)

### **4 - HARDWARE NECESSÁRIO**

Para rodar o Conectiva Linux 4.0 (CL 4.0) é necessário processador 386 ou superior, 100 a 150Mb de espaço livre em disco, 16Mb de RAM (pra ficar leve), CD-ROM e unidade de disquete 3 1/2. O produto suporta a maioria dos computadores, controladoras SCSI e IDE, placas de som, impressoras, placas de rede Ethernet, scanners e demais periféricos do mercado.

### **5 - ENTRANDO E SAINDO DO SISTEMA**

Chegamos ao momento em que todos queriam: ver a cara do Linux. Antes o veremos no modo texto, e mais a frente veremos a interface gráfica.

Ligue a máquina e aguarde a inicialização do sistema.

A primeira vez que você acessar o sistema Conectiva Linux, o acesso deverá ser realizado com o superusuário **root**. Este é o nome da conta que tem acesso completo a todos os componentes do sistema. Normalmente, a conta de superusuário é somente utilizada na execução de tarefas de administração do sistema, como a criação de novas contas, desligar o sistema, etc. Isso se deve ao fato de que o acesso irrestrito do superusuário quando mal

utilizado poderá provocar grandes estragos ao sistema. Então seja cuidadoso ao acessar o sistema como **root**, e use a conta de superusuário somente quando realmente for necessário. Para o acesso inicial, informe **root** na linha de comando **login:** . Pressione [Enter]. Uma linha de comando **password:** aparecerá. Digite sua senha e pressione [Enter]. Então deverá surgir algo como:

```
[root@guarani /root] #
```

Caso o usuário ou a senha estejam mal informadas, tente novamente até que a linha acima apareça.

Agora...meus parabéns...Você conseguiu acessar o sistema com sucesso. Próximo passo: aprendendo a sair do sistema.

Apesar de muitos interpretadores de comandos terem uma instrução **logout** ou **exit**, muitos usuários simplesmente teclam **[Ctrl]-[D]**. Isso deve retornar à linha de acesso ao sistema. Mas não é só isso...

Ainda que isso seja um pouco mais complexo que simplesmente desligar o botão de energia, o encerramento do Linux tem alguns detalhes adicionais. Uma vez que você esteja pronto para desligá-lo, não quer dizer que o sistema esteja apto para tal. Para melhor entender o que queremos dizer, execute o comando:

```
ps ax
```

Cada uma das linhas listadas pelo comando **ps** representa um processo em execução. Cada processo pode estar trabalhando com arquivos, e caso o sistema seja simplesmente desligado, esses processo não terão a chance de fechar todos os arquivos e finalizarem a execução de maneira correta. Logo para poder desligar o sistema corretamente, é necessário avisar aos processos que finalizem normalmente sua execução. Para tanto, pode-se usar o comando **shutdown**. Este comando pode ser executado somente pelo superusuário e será necessário acessar o sistema como tal ou executar o comando **su** (ver mais adiante) para tornar-se superusuário **root**. A sintaxe básica do **shutdown** é:

```
Shutdown <options> <time>
```

Nota: o programa **shutdown** reside no diretório `/sbin`. Caso sua variável de ambiente `PATH` não inclua `/sbin`, será necessário fornecer o seu caminho completo como parte do comando (por exemplo, `/sbin/shutdown -h now`).

Em muitos casos, pode-se incluir uma das seguintes opções:

- h suspende o sistema quando a finalização estiver completa (halt)
- r reinicializa o sistema quando a finalização estiver completa (reboot)

Caso não seja incluída nenhuma das opções, **shutdown** reinicializará o sistema em modo monousuário. A menos que esteja claro em porque usar o sistema em modo monousuário, esta opção não deve ser utilizada. Simplesmente informe o comando **shutdown**(desta vez com `-h` ou `-r`) e ele finalizará normalmente. O comando fornece ainda grande flexibilidade em termos de tempo. Caso se deseja que seja executado imediatamente, simplesmente informe a palavra **now**. Caso se deseje que o sistema seja desligado em cinco minutos a partir de agora, basta informar **+5**. Em assim sendo, o comando

```
Shutdown -r +15
```

Significa “desligue o sistema em quinze minutos a partir de agora e reinicialize após o encerramento ter sido completado”. **Shutdown** tem diversas outras opções disponíveis, sendo



que descreveremos aqui somente o básico necessário para executar tarefas simples de desligamento. Para aprender mais, execute o comando **man shutdown** para saber mais características deste comando. **Em português !!!**

## 6 - AMBIENTE DE TRABALHO (SHELL)

O **shell** é um interpretador de comandos. Ele interage com o sistema operacional, fazendo a ligação entre os comandos que você digita e as atividades que o kernel pode realizar. O propósito do **shell** é tornar o sistema operacional mais amigável, por ser muito mais fácil para o usuário lembrar o nome do comando que o conjunto de chamadas de sistema que estão por trás dele. O Linux possui mais do que um **shell**, cada qual com características ligeiramente diferentes e funções especiais. Você pode usar o **shell** que preferir. O **shell** padrão para o Linux é o Bourne Again Shell (*bash*). Entre outros, tem o C-shell (*cs*h), Korn shell (*ksh*), etc.

Todos os shells do Linux fazem a diferenciação entre letras maiúsculas e minúsculas. Isto significa que os nomes de arquivos devem ser especificados como estão, usando só letras e caracteres maiúsculos, só minúsculos ou misturados. Os nomes de arquivos podem ter até 256 caracteres e podem conter muitos tipos diferentes de caracteres. Por exemplo, os nomes de arquivos a seguir são exemplos válidos:

```
Um_nome_arquivo_longo+uma_longa_extensão
Arquivoqueélongodema isparaamaior iadosnomesdearquivos
PagueaContaouNuncaMais VeráSeuGatodeNovo
```

Outro recurso do shell padrão é a finalização de comandos. Para digitar rapidamente ou localizar o nome de um comando ou todos os comandos com grafias semelhantes, digite as primeiras letras do nome de um comando e, depois, pressione a tecla <TAB>. Por exemplo, digite o seguinte:

```
[root@guarani /root]# contr
```

Pressione a tecla <TAB> e ficará...

```
[root@guarani /root]# control-panel
```

Se você digita somente as duas primeiras letras, precisa pressionar duas vezes a tecla <TAB>. Por exemplo, digite o seguinte:

```
[root@guarani /root]# pi
```

Depois, pressione duas vezes a tecla <TAB> para obter os seguintes resultados:

```
Piltoppm  pic    pick    picttoppm  pilot  ping
Pi3topbm  pic2tpic  pico    pidof    pine
```

O shell exibe todos os arquivos cujos nomes contêm as letras especificadas.



**Shell Script** - O shell possibilita a interpretação tanto de comandos digitados quanto de comandos armazenados em arquivos, denominados **shell script**, que não são nada mais do que um arquivo texto com permissão de execução. No Windows, é conhecido como arquivo de lote (batch). Para criar o shell script, crie um arquivo de texto e ajuste suas permissões para que ele se torne executável. Este arquivo pode ser criado com um editor como o joe ou simplesmente o redirecionamento para um arquivo da saída de um comando `cat`. **Nota:** Para tornar um arquivo texto em um shell script deve-se usar o comando `chmod` para mudar suas permissões de acesso.

## 07 - CONTAS E SENHAS

Conforme mencionado anteriormente, não é aconselhável utilizar a conta de superusuário todo o tempo. Inevitavelmente um erro será cometido, e a checagem de acesso que normalmente evita esse tipo de erro, não funcionará, uma vez que ao superusuário é permitido fazer qualquer coisa no sistema. Bem ser você não deve acessar o sistema como superusuário, com que nome deverá acessá-lo? Com o seu nome, obviamente. Para isso, você precisa saber como criar contas no seu Linux.

**CONTAS** – Assim que ligado, o sistema oferece diversas formas de criar novas contas. Usaremos inicialmente o método mais básico: o comando **useradd**. Basicamente, tudo que se deve fazer é informar (como superusuário):

```
[root@guarani /root] # useradd aluno
```

```
[root@guarani /root] #
```

Foi muito simples, não ? Bem, vamos acessar o sistema:

```
Conectiva Linux Versão 4.0 (Guarani)
Kernel 2.2.5 em um i486
login: aluno
Password:
Login incorrect
```

```
login:
```

No campo de senha do usuário “aluno”, simplesmente pressione [Enter]. Bem, essa não é uma senha muito indicada. Vejamos então como especificar uma senha para uma conta nova.

**SENHAS** – O comando **passwd** pode ser usado para:

- Especificar senhas para contas recém-criadas;
- Mudar as senhas de contas já existentes;
- Mudar a senha de contas com as quais se esteja acessando o sistema.

As primeiras duas situações são realmente as mesmas; não há realmente uma diferença entre uma conta que já exista e uma que acabou de ser criada. Tudo o que você deve saber é que deve acessar o sistema como superusuário (root), e então especificar o

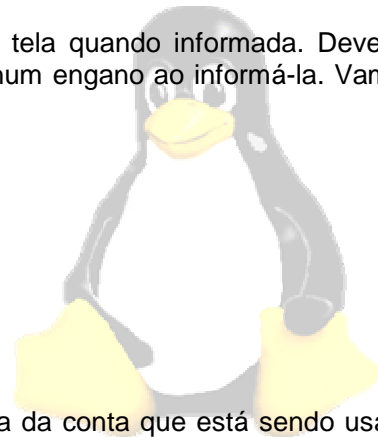
nome da conta cuja senha se deseja alterar. Usando a conta que acabamos de criar, temos o seguinte exemplo:

```
[root@guarani /root] # passwd aluno
New Unix password:
Retype new UNIX password:
passwd: all authentication tokens update sucessfully
[root@guarani /root] #
```

Como se pode perceber, a senha não aparece na tela quando informada. Deve-se ainda digitar duas vezes, para garantir que não houve nenhum engano ao informá-la. Vamos acessar o sistema com a conta recém-criada novamente:

```
Conectiva Linux Versão 4.0 (Guarani)
Kernel 2.2.5 em um i486
login: aluno
Password:
```

```
[root@guarani /root] #
```



Uma vez dentro do sistema, pode-se alterar a senha da conta que está sendo usada utilizando-se o comando **passwd** sem o nome da conta. Neste caso, ele solicitará a senha atual da conta, seguida do comando de nova senha. Por exemplo, para uma conta chamada aluno, teremos:

```
[root@guarani aluno] # passwd
Changing password for aluno
(current) UNIX password:
Retype new UNIX password:
passwd: all authentication tokens update sucessfully
```

```
[root@guarani /root] #
```

É moleza!!!

**COMANDO SU** – Há momentos em que pode ser necessário processar um ou dois comandos como outro usuário. É normal que administradores de sistema tenham esse tipo de demanda – eles (como todos os bons administradores de sistemas) usam a sua conta pessoal e sem privilégios especiais a maior parte do tempo. Mas caso uma senha de usuário necessite ser alterada ou as permissões de um determinado arquivo devam ser ajustadas, isso pode não ser possível com uma conta de usuário simples. Tais tarefas não levam mais que um minuto, e pode ser um tanto aborrecido ter e sair e entrar no sistema diversas vezes, somente para executar pequenas tarefas como superusuário. Uma abordagem mais simples consiste na utilização do comando **su**. Com este comando, a sessão atual pode transformar-se na sessão do superusuário ou outro usuário. No seguinte exemplo, o usuário “aluno” decide tornar-se superusuário:

```
[root@guarani aluno]$ su
Password:
[root@guarani aluno]#
```

Como é possível perceber através do comando **su**, o usuário comum obtém os poderes de superusuário, após a informação da senha do **root**. Mas caso se observe mais atentamente, notará que o indicador de linha que tinha um sinal (\$), passa a ter (#). Esta é a forma tradicional de indicar se um ambiente de trabalho está sendo executado como superusuário ou não.

É possível ainda tornar-se outro usuário. Pode-se fazer isso, sem informar a senha (no nosso exemplo a senha do usuário não seria requerida) através do comando **su conta\_do\_usuario**, caso se esteja utilizando o superusuário, ou informando a senha do usuário que se deseje alterar. Você achará o comando **su** muito útil, particularmente se está agindo como deve um administrador do sistema.

Para excluir um usuário usa-se o comando **userdel** :

Exemplo:

```
[root@guarani /root]# userdel -r aluno
```

Lembrando que somente o superusuário poderá fazê-lo.

## 08 - CONSOLES VIRTUAIS

O Linux é um sistema multi-tarefa, por isso, ele pode ser acessado por vários consoles ao mesmo tempo, assim como pode ser rodado vários programas ao mesmo tempo. Para mudar o console do 1 a 6, utilize:

ALT+N (Onde N é o número do console)

Exemplo:

ALT+1, ALT+2, ALT+3, ALT+4, ALT+5, ALT+6

Agora você pode ir para o próximo console e o antecedente com:

ALT+RIGHT (Vai pra 1 console A FRENTE)  
ALT+LEFT (Vai pra 1 console ATRÁS)

Se você quiser ir para outra sessão em sair do console, utilize o comando **su**

## 09 - COMANDOS BÁSICOS

Para facilitar as coisas, listaremos os comandos iniciando pelos mais usados (fáceis), e dificultando à medida em que não houver nenhuma dúvida. Facilitando mais ainda, em alguns comandos do Linux colocamos o comando equivalente no DOS.

Não se esqueça que eles devem ser digitados com letras minúsculas.  
Começando pelo...

<b>COMANDO</b>	<b>FUNÇÃO</b>	<b>DOS</b>
<b>LS</b>	Lista os arquivos do diretório	<b>DIR</b>
<b>RM</b>	Remove arquivos	<b>DEL</b>
<b>CP</b>	Copia arquivos	<b>COPY</b>
<b>MV</b>	Renomeia arquivos e diretórios	<b>RENAME</b>
<b>MV</b>	Move arquivos e diretórios	<b>MOVE</b>
<b>CAT</b>	Mostra o conteúdo do arquivo	<b>TYPE</b>
<b>MORE</b>	Mostra o conteúdo do arquivo, paginado	<b>MORE</b>
<b>PWD</b>	Mostra o diretório corrente	-
<b>MKDIR</b>	Cria diretório	<b>MD</b>
<b>RMDIR</b>	Apaga diretório	<b>RM</b>
<b>CD</b>	Navega entre os diretórios	<b>CD</b>
<b>CLEAR</b>	Limpa a tela	<b>CLS</b>
<b>DF</b>	Informa os dados de ocupação do sistema de arquivos	<b>CHKDSK</b>
<b>FREE</b>	Informa como está sendo utilizado a memória	<b>MEM</b>
<b>FIND</b>	Procura arquivos	<b>DIR /S</b>
<b>WHO</b>	Informa os usuários conectados e os respectivos terminais	-
<b>WHOAMI</b>	Mostra quem você é	-
<b>EXIT</b>	Sai da sessão atual	-
<b>LOGOUT</b>	A mesma coisa...	-
<b>CAL</b>	Exibe um calendário	-
<b>DATE</b>	Retorna a data e a hora	<b>DATE</b>
<b>DU</b>	Informa o espaço ocupado pelos arquivos ou diretórios	-
<b>FORMAT</b>	Formata um floppy disk	<b>FORMAT</b>
<b>KILL</b>	Termina um processo	-
<b>PS</b>	Exibe um status dos processos	-
<b>VI</b>	Editor de tela cheia	-
<b>WC</b>	Exibe detalhes no tamanho do arquivo	-
<b>MAN</b>	Exibe a ajuda de comandos	<b>HELP</b>
<b>UNAME</b>	Informa os dados do sistema	-
<b>UPTIME</b>	Informa há quanto tempo foi dado boot	-
<b>ADDUSER</b>	Adiciona usuários ao sistema	-
<b>USEDEL</b>	Apaga usuários no sistema	-
<b>PASSWD</b>	Altera senha do usuário no sistema	-
<b>ARCH</b>	Informa a arquitetura do computador	-

<b>SHUTDOWN</b>	Desliga o sistema	-
<b>LESS</b>	Melhor que o MORE	<b>MORE</b>
<b>CHMOD</b>	Altera a permissão dos arquivos	-
<b>LPD</b>	Imprime no modo texto	<b>PRINT</b>
<b>CPIO</b>	Ferramenta para backup (tipo TAR)	-
<b>LHA</b>	Compactador de arquivos	<b>LHA</b>
<b>WRITE</b>	Envia mensagens p/ outro usuário na rede (BROADCAST - Novell)	-
<b>MOUNT</b>	Permite acesso aos dados de unidades de disco	-

## 10 - EDITANDO NO LINUX (MODO TEXTO)

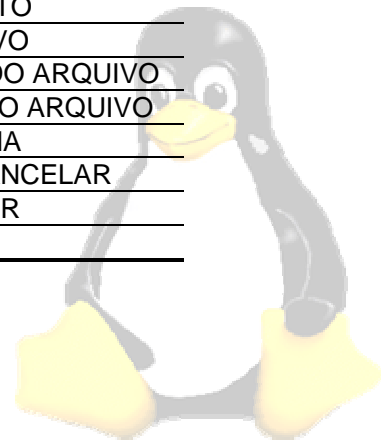
Para editarmos arquivos em modo texto, usaremos o editor **joe**, devido a grande facilidade de uso. Os seus comandos internos são muito parecidos com nosso velho WordStar (ou WS). Além do joe, há ainda os editores **vi**, **jed**, **emacs**, e outros pouco divulgados.

Sintaxe:

```
[root@guarani /root]# joe arquivo
```

Os comandos internos mais usados são:

COMANDO	FUNÇÃO
^KF	PROCURAR TEXTO
^KB (BEGIN)    ^KK (END)	MARCAR TEXTO
^KC	COPIAR
^KM	MOVER
^KY	DELETAR TEXTO
^KD	SALVA ARQUIVO
^KU	VAI P/ INICIO DO ARQUIVO
^KV	VAI P/ FINAL DO ARQUIVO
^Y	DELETAR LINHA
^C	ABORTAR / CANCELAR
^KX	GRAVAR E SAIR
^KH	HELP



## 11 - USANDO COMPACTADORES

Para compactar arquivos, usaremos o **gzip**, que é distribuído nos termos da GNU. Existem outros compactadores, como o **unzip**, **zip**, **compress**, **bzip**, **unrarj**, e muitos outros que funcionam da mesma maneira que no formato DOS, mas a maioria dos arquivos pra Linux na Internet são compactados com o **gzip**, acompanhado do **tar**, que também abordaremos agora.

Suponhamos que você tenha o seguinte arquivo em um diretório:

```
[root@guarani /root]# ls -l
-rwxr-xr-x  2  root  users  16000  Nov  04  14:00  teste

[root@guarani /root]# gzip teste
[root@guarani /root]# ls -l
-rwxr-xr-x  2  root  users    900  Nov  04  14:01  teste.gz
```

Note que o antigo arquivo, teste, foi deletado depois que o gzip passou sem problemas.

Geralmente os arquivos, quando compactados no Linux, mantém a mesma extensão.

Ex.: teste.txt (antes)  
Teste.txt.gz (depois)

Para se descompactar um arquivo \*.z ou \*.gz, usa-se o comando **gunzip**, da seguinte forma:

```
[root@guarani /root]# ls -l
-rwxr-xr-x  2  root  users    900  Nov  04  14:01  teste.gz

[root@guarani /root]# gunzip teste.gz
[root@guarani /root]# ls -l
-rwxr-xr-x  2  root  users  16000  Nov  04  14:01  teste
```

Note-se também que o antigo arquivo compactado foi deletado. O **gzip** não funciona igual ao pkzip e winzip quando se trata de diretórios. Para se compactar múltiplos arquivos e diretórios diferentes, usa-se a associação do comando **tar** com o comando **gzip**.

O **tar** é um comando designado para criação de arquivos que contêm diversos outros arquivos internamente. Usado muito na criação de backup's em programas e sistemas empresariais devido a sua característica de poder unir arquivos e diretórios em apenas um arquivo. Nas empresas que usam o tar, geralmente usam a própria data do dia como nome para os arquivos de backup. Exemplo:

```
[root@guarani /backup]# ls -l
-rwxr-xr-x  2  root root  11524211  Dez  29 14:01  bkp29121998
```

Seu uso, bastante simples, é feito da seguinte forma:

```
tar <opções> <arquivos>
```

#### OPÇÕES :

- c cria um novo arquivo tar.
- t lista o conteúdo do arquivo tar.
- x extrai o conteúdo do arquivo tar.
- v mostra na tela os arquivos que estão sendo compactados/descompactados
- f arquivo de origem/destino.
- z filtra o arquivo com gzip.
- r para colocar novos arquivos no final do arquivo .tar
- u para fazer um update nos arquivos do arquivo .tar
- d para comparar o conteúdo do arquivo com arquivos do sistema
- k para manter arquivos existentes no diretório.

Imagine, por exemplo, que você tenha um diretório chamado “teste” com os seguintes arquivos:

```
[root@guarani /teste]# ls -l
-rwxr-xr-x  1  root root      21    Nov  08    16:06  Makefile
-rwxr--r--  1  root root     847   Dez  11    04:41  README
-r-xr-xr-x  1  root root    9254   Mar  16    12:55  teste.txt
-rw-rw-rw-  1  root root      36    Jul  01    09:56  alysson.net
-rwxrwxrwx  1  root root     395   Nov  12    11:26  lixo
-rwxr-xr-x  1  root root    1223   Mai  29    06:01  profile
```

Para que compactemos este diretório, é necessário que antes de tudo, criar um arquivo .tar

```
[root@guarani /]# tar cvf teste.tar teste
teste/
teste/Makefile
teste/README
teste/teste.txt
teste/alysson.net
teste/lixo
teste/profile
```

```
[root@guarani /]# ls -l
-rwxr-xr-x  1  root root    11776   Nov  18    16:52  teste.tar
```

O comando digitado irá juntar todo o conteúdo do diretório em um único arquivo .tar. A opção “v” é para que possamos ver o que está sendo colocado no arquivo teste.tar.

Note que esse novo arquivo contém o número somado de bytes dos arquivos que estavam no diretório teste.

Para se compactar este arquivo, usa-se o comando gzip da seguinte forma:

```
[root@guarani /]# gzip -9 teste.tar
[root@guarani /]# ls -l
-rwxr-xr-x  1  root root      921    Nov  18    16:54  teste.tar.gz
```

O inconveniente disso, é que temos que fazer o tar e depois compactá-lo. Para economizar tempo, usa-se a seguinte linha de comando:

```
[root@guarani /]# tar cvf teste.tar teste | gzip > teste.tar
[root@guarani /]# ls -l
-rwxr-xr-x  1  root root      921    Nov  18    16:54  teste.tar.gz
```

Para se descompactar um arquivo dessa forma, usa-se o comando:

```
[root@guarani /]# tar xvfz teste.tar.gz
[root@guarani /]# ls -l
-rwxr-xr-x  1  root root      21    Nov  08    16:06  Makefile
-rwxr--r--  1  root root     847   Dez  11    04:41  README
-r-xr-xr-x  1  root root    9254   Mar  16    12:55  teste.txt
-rw-rw-rw-  1  root root      36    Jul  01    09:56  alysson.net
-rwxrwxrwx  1  root root     395   Nov  12    11:26  lixo
-rwxr-xr-x  1  root root    1223   Mai  29    06:01  profile
```



Outro Exemplo :

```
[marisa@guarani /tmp]$ l -d textos
drwxrwxr-x  4 marisa marisa      1024 jul 23 11:44 textos/

[marisa@guarani /tmp]$ tar cvzf textos.tgz textos
textos/
textos/perl/
textos/java/

[marisa@guarani /tmp]$ l textos.tgz
-rw-rw-r--  1 marisa marisa      155 jul 29 11:24 textos.tgz

[marisa@guarani /tmp]$ tar tvzf textos.tgz
drwxrwxr-x marisa/marisa  0 1998-07-23 11:44 textos/
drwxrwxr-x marisa/marisa  0 1998-07-23 11:38 textos/perl/
drwxrwxr-x marisa/marisa  0 1998-07-23 11:38 textos/java/
```

## 12 - CONTROLE DE ACESSO DE ARQUIVOS

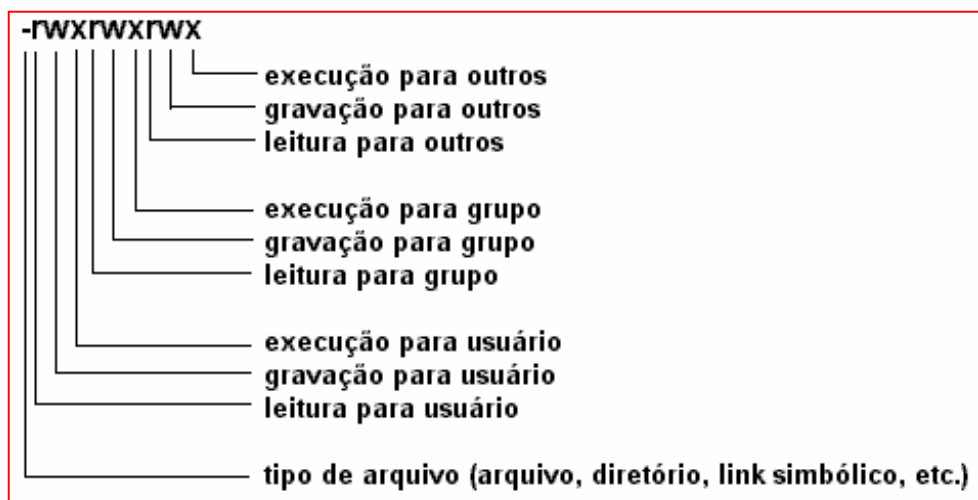
### ❖ Conceito de Permissões

Num certo momento, você deve ter pensado: “ – O que significa esses rwx-wx- na frente dos arquivos ?” Bom...essas são as permissões de arquivos para usuários. Tanto no Unix como no Linux, podemos (superusuário) controlar quem pode ler, alterar, apagar, gravar e executar nossos arquivos. As permissões são classificadas quanto ao nível do usuário, grupo e outros. Cada arquivo tem um UID (User ID), e um GID (Group ID), que mostra quem é o dono do arquivo. Segue o exemplo:

```
[root@guarani /]# ls -l
-rwxr-xr--  1 jose operador    21   Nov  08 16:06 .profile
```

O arquivo **.profile** tem 21 bytes, o dono do arquivo é o usuário José, que pertence ao grupo operador.

As permissões são divididas da seguinte forma:



Todo arquivo determina quais usuários têm acesso a ele e com que finalidade. Cada categoria de usuários possui um conjunto distinto de permissões de acesso ao arquivo. Cada conjunto de permissões de acesso significa presença ou ausência de permissões para: leitura (r); escrita (w); execução (x).

Cada usuário do sistema possui três conjuntos (rwx) de permissão para cada arquivo. O sistema de permissões dá ao usuário mais segurança, pois permite que ele tenha um maior controle ao acesso de seus arquivos e diretórios. Isto dá mais segurança não só ao usuário, mas a todo o sistema.

Para o arquivo acima, o dono possui permissão de leitura(r), escrita(w) e execução(x). O grupo possui permissão de leitura(r) e execução(x). Outro usuário que não o dono e não pertencente ao grupo do dono possui permissão somente de leitura(r).

#### ❖ Mudando Permissões

**chgrp** – Este comando muda o grupo dos arquivos/diretórios dados como argumento. O parâmetro *group* pode ser tanto um número (gid - identificador de grupo), como um nome de grupo encontrado no arquivo de grupos do sistema /etc/group. O usuário deve ser membro do grupo especificado e dono do arquivo (ou o superusuário). O formato do comando é simplesmente:

**chgrp <novo grupo> <arquivo>**

**chown** - Muda o dono dos arquivos e diretórios para um novo dono, que pode ser um nome de acesso ou a identificação de usuário (número associado ao nome do usuário). Nota: o comando chown pode tanto mudar o dono dos arquivos, bem como o próprio grupo de arquivos. Formato:

**chown <id do novo usuário> <arquivo>**

**chmod** – O comando chmod é usado para alterar as permissões associadas ao objeto. O que você realmente está fazendo é alterar o modo do arquivo. Existem duas maneiras de especificar as permissões do objeto. Você pode usar o sistema de código numérico ou o sistema de codificação alfabético. Usando o sistema alfabético, usaremos as letras **u**(usuário), **g**(grupo), **o**(outros) e **a**(all > todos). As permissões podem ser alteradas usando os sinais **+**(mais) e **-**(menos). Por exemplo, para acrescentar as permissões **ler** e **executar** ao proprietário e grupo do arquivo **.profile**, usaremos o comando da seguinte forma:

```
[root@guarani /]# ls -l
--w-w-rwx  1  jose operador    21   Nov   08   16:06  .profile

[root@guarani /]# chmod ug+rx .profile
[root@guarani /]# ls -l
-rwxrwxrwx  1  jose operador    21   Nov   08   16:06  .profile
```

Para remover as permissões **ler** e **executar** de usuário e de grupo do arquivo **.profile**, simplesmente iremos trocar o sinal de **+**(mais) pelo de **-**(menos):

```
[root@guarani /]# chmod ug-rx .profile
[root@guarani /]# ls -l
--w-w-rwx  1  jose operador    21   Nov   08   16:06  .profile
```

Usando o sistema de codificação numérico, você tem sempre de informar o valor absoluto da permissão, independente das suas permissões prévias. O sistema numérico é

baseado em três conjuntos de números de base dois. Existe um conjunto para cada categoria de usuário, grupo e outros. Os valores são 4(ler), 2(gravar) e 1(executar). Esses valores são acrescentados juntos para fornecer o conjunto de permissões para aquela categoria. Com o sistema numérico você sempre especifica todas as três categorias. Portanto, para que o proprietário do arquivo .profile tenha permissões de ler, gravar e executar o arquivo e ninguém mais tenha qualquer permissão, você deve usar o valor 700, da seguinte forma:

**chmod 700 .profile**

Para tornar o mesmo arquivo legível e gravável para o usuário e legível pelo grupo e outros, você segue a mesma lógica matemática: para o primeiro conjunto de permissões, o usuário, o valor para legível é 4, e o valor para gravável é 2. A soma desses dois é 6. O próximo conjunto de permissões, o grupo, é read-only, portanto é 4. As definições para os outros, assim como o grupo, é 4. Portanto o comando seria **chmod 644 .profile**.

### 13 – CD-ROM's, Disquetes, Discos Rígidos e Sistemas de Arquivos – Visão Geral

Um sistema de arquivos é composto por arquivos e diretórios, iniciando em um único diretório denominado raiz. Este diretório pode conter qualquer número de arquivos ou de diretórios, com cada diretório por sua vez seguindo o mesmo conceito e padrões. Um sistema de arquivos padrão normalmente se parece com uma árvore invertida, com os diretórios como galhos e os arquivos como folhas. Sistemas de arquivos residem em unidades de armazenamento de massa como disquetes, discos rígidos e CD-ROMs.

Por exemplo, uma unidade de disquetes no DOS ou Windows é normalmente referenciada como A:. Isso descreve o dispositivo (A:) e o diretório raiz do dispositivo. O disco rígido primário, em sistemas similares, é tipicamente referenciado como C uma vez que a especificação de dispositivos para o primeiro disco rígido é C:. Para especificar o diretório raiz do dispositivo C, pode-se utilizar C:.

Neste caso, teremos então dois sistemas de arquivos - um em A: e o outro em C:. Para especificar qualquer arquivo em um sistema de arquivos DOS/Windows, deve-se especificar o dispositivo no qual ele reside, ou ele deve residir no dispositivo padrão do sistema (o qual é a origem do indicador DOS de linha de comando - é o dispositivo padrão em um sistema com uma única unidade de disco rígido).

Sob Linux é possível definir sistemas de arquivos residentes em diferentes meios de armazenamento como se fossem um único e grande sistema de arquivos. Isso pode ser feito através da definição de um dispositivo dentro de um sistema de arquivos. Por exemplo, enquanto um sistema de arquivos de um diretório raiz de um disquete em DOS pode ser referenciado como A:, o mesmo dispositivo pode ser acessado no Linux com um diretório denominado, por exemplo como /mnt/floppy.

O processo de mesclar sistemas de arquivos desta forma é conhecido como montagem. Quando um dispositivo está montado significa que ele pode ser acessado pelos usuários do sistema. O diretório através do qual o sistema de arquivos pode ser acessado é conhecido como ponto de montagem. No exemplo anterior, /mnt/floppy era o ponto de montagem do disquete. Note que não há restrições (além das convenções normais) de nome de pontos de montagem. Poderíamos facilmente denominar o ponto de montagem com /longo/caminho/para/a/unidade/de/disquete ou simplesmente /A. Um ponto a ser lembrado é que todos os diretórios e arquivos de um dispositivo têm a sua localização no sistema relacionada com o ponto de montagem. Por exemplo, consideremos a seguinte configuração:

- **Um sistema Linux**
  - / - diretório raiz do sistema
  - /cnc - ponto de montagem do CD-ROM

- **Um CD-ROM**

- / - diretório raiz do CD-ROM
- /imagens - um diretório de imagens do CD-ROM
- /imagens/antigas - um diretório de imagens antigas do CD-ROM

Enquanto que na descrição acima, temos uma apresentação individualizada dos sistemas de arquivos e ao se montar o CD-ROM em /cnc, a nova estrutura de diretórios do sistema terá a seguinte configuração:

- **Um sistema Linux com o CD-ROM montado:**

- / - diretório raiz do sistema
- /cnc - diretório raiz do CD-ROM
- /cnc/imagens - um diretório de imagens do CD-ROM
- /cnc/imagens/antigas - um diretório de imagens antigas do CD-ROM

Para montar um sistema de arquivos esteja seguro de estar acessando o sistema como superusuário ou de usar o comando `su` (`man su` - em português). Uma vez tendo os privilégios de superusuário, execute o comando `mount` (`man mount` - em português) seguido pelo dispositivo e pelo ponto de montagem. Por exemplo, para montar a primeira unidade de disquete em `/mnt/floppy`, pode-se digitar o seguinte comando:

```
[root@guarani /root]# mount /dev/fd0 /mnt/floppy
```

Para acessar os dados em um disquete formatado em **ext2**, basta digitar `cd /mnt/floppy`. Na instalação o *Conectiva Linux* irá criar um arquivo chamado `/etc/fstab`. Este arquivo contém informações que permitem sintetizar os comandos de montagem de dispositivos. Usando-se as informações contidas naquele arquivo, pode-se comandar somente `mount` e então, ou o ponto de montagem ou o dispositivo. O comando `mount` irá então procurar o restante das informações em `/etc/fstab`. É possível modificar manualmente o arquivo com um editor de texto simples, ou pelo utilitário **Linuxconf**, dentro do Sistema X

- **Revendo:**

Para acessar um floppy disk, isto é, um disco flexível, você terá que utilizar o comando `mount`. Você terá que ter o driver e o device respectivamente (`fd0`, `fd1`, `fd2`, etc.). Então você deverá digitar:

```
mount /dev/fd0 /diretório_ao_disco_ser_acessado
```

Seguindo o formato acima, coloque um disquete (3 1/2") no drive e digite:

```
[root@guarani /root]# mount /dev/fd0 /mnt
```

Isto fará com que você acesse o disquete que está no drive atualmente.

Quando você quiser retirar o disco geralmente deve-se 'desmontá-lo' primeiro. Digite:

```
[root@guarani /root]# umount /diretório_acessado
```

Se preferir, pode também criar um shell script, que pode se chamar, por exemplo de 'diskon' (Para ativar) e 'diskoff' (Para desativar), ou os nomes que preferir. Então para melhor utilização, coloque este arquivo em um diretório PATH. Ou até mesmo, especifique-o com o comando `alias` dentro do arquivo de configuração do shell padrão.

Para acessar o CD-ROM, a forma e os comandos são os mesmos, alterando apenas o device que corresponda ao drive desejado (neste caso, `/dev/cdrom`)

Se o comando `mount` é executado sem parâmetros, ele lista todos os sistemas de arquivos atualmente montados.

Para maiores esclarecimentos, use o comando `man (man mount)`.

## 14 – USANDO LINUX E WINDOWS NO MESMO DISCO RÍGIDO.

Existem catalogados na Internet mais de cinco mil softwares para plataforma Linux, dentre editores de texto, utilitários gráficos, navegadores, utilitários para internet, banco de dados, programas de áudio, ferramentas de programação, games, etc. e muitos outros softwares estão em fase de migração. Enquanto eles não chegam, você pode optar em usar dois sistemas operacionais em sua máquina: Linux e Windows.

Aí você pergunta: “E pode ???” Pode...Para isso, você tem que ter instalado o **LILLO** no Linux.

O LILLO é um gerenciador de boot, onde você escolhe qual sistema operacional deseja entrar ao ligar a máquina.

Quando você liga a máquina, o Linux mostra a seguinte mensagem:

```
LILLO boot:
```

É nesta hora que você define qual o sistema deseja “bootar” digitando o nome do sistema na frente.

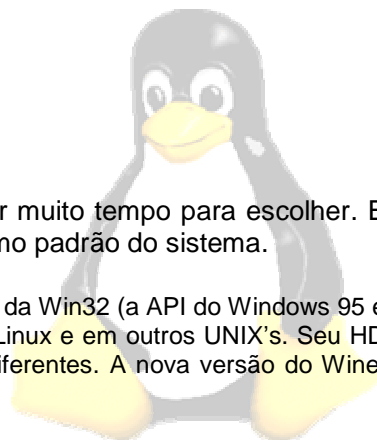
```
LILLO boot: windows  
ou
```

```
LILLO boot: linux
```

Dependendo da configuração feita no LILLO, você pode não ter muito tempo para escolher. E se demorar muito, o sistema abrirá naquele que foi definido como padrão do sistema.

**WINE** - O Wine é uma implementação free da API do Windows 3.x e da Win32 (a API do Windows 95 e posteriores) que permite a execução de programas do Windows no Linux e em outros UNIX's. Seu HD tem de ter instalado tanto o Linux como o Windows em partições diferentes. A nova versão do Wine, apelidada de Hallowine, já está disponível:

<ftp://tsx-11.mit.edu/pub/linux/ALPHA/Wine/development/Wine-991114.tar.gz>

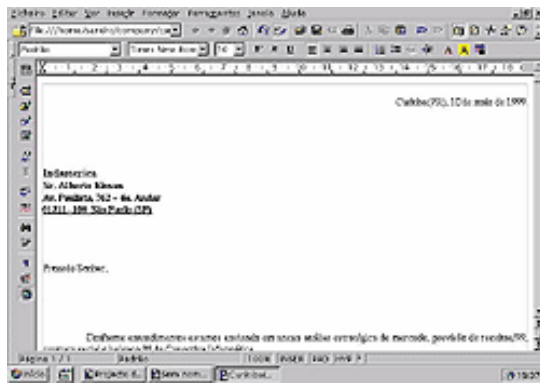


## 15 – INTERFACES GRÁFICAS.

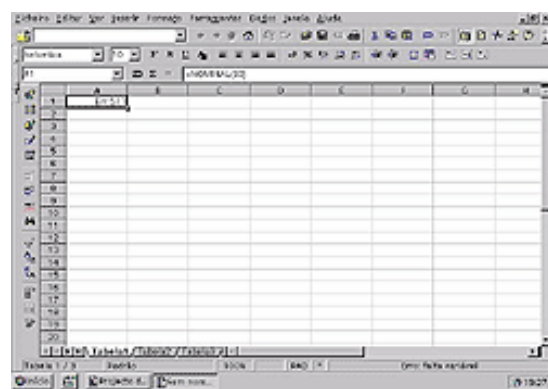
Dentre as interfaces gráficas mais usadas estão a X-Window, KDE e Gnome. Todas podem ser instaladas de uma vez no Linux durante a instalação, mas somente uma pode ser usada de cada vez.

Você pode configurar seu sistema X para inicializar pelo boot, ou acessando-o pelo modo texto com o comando equivalente, de acordo com o modo como vai usar o sistema.

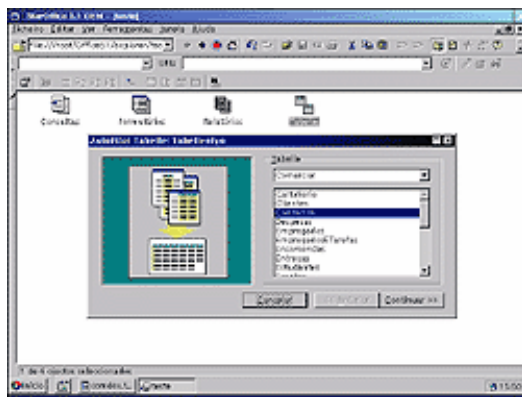
**Editor de texto**



**Planilha Eletrônica**



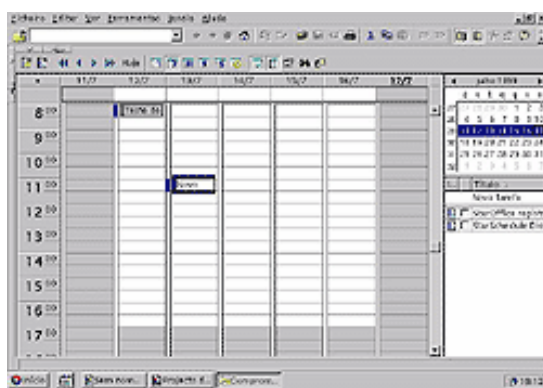
**Banco de Dados**



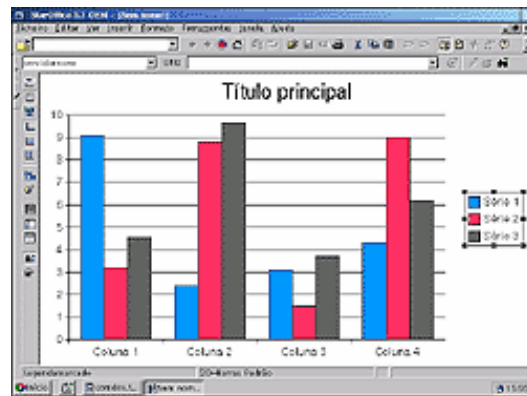
**Editor de Apresentações**



**Agenda**



**Editor de Gráficos**





## 16.2 - Sites (em inglês)

<http://www.linux.org> - Home page da Linux Organization, site oficial do Linux  
<http://filewatcher.org/>  
<http://www.linuxgazette.com/>  
<http://www.linuxjournal.com/>  
<http://www.linuxapps.com/>  
<http://www.linuxberg.com/>  
<http://www.linux-2000.org/>

## 16.3 - Servidores de FTP

<ftp.iis.com.br> - diversos arquivos da internet.  
<ftp.conectiva.com.br> - arquivos variados da Conectiva  
<ftp.versatec.com> - contém diversos softwares para Linux.  
<ftp.ibp.fr> : /pub/linux (França)  
<ftp.cc.gatech.edu> : /pub/linux (EUA - sudeste: Suranet)  
<ftp.cdrom.com> : /pub/linux (EUA)  
<ftp.informatik.tu-muenchen.de> : /pub/comp/os/linux (Alemanha)  
<ftp.ibr.cs.tu-bs.de> : /pub/os/linux (Alemanha)  
<ftp.dfv.rwth-aachen.de> : /pub/linux (Alemanha)  
<ftp.informatik.rwth-aachen.de> : /pub/Linux (Alemanha)  
<ftp.cc.monash.edu.au> : /pub/linux (Austrália)  
<ftp.dstc.edu.au> : /pub/Linux (Austrália: Queensland)  
<ftp.sun.ac.za> : /pub/linux (África do Sul)  
<ftp.inf.utfsm.cl> : /pub/Linux (Chile)  
<ftp.zel.fer.hr> : /pub/Linux (Croácia)  
<lcmi.ufsc.br> : /pub/diversos/linux (Brasil : Santa Catarina) Slackware  
<cesar.unicamp.br> : /pub3/linux (Brasil : São Paulo) Slackware  
<ftp.ime.usp.br> : /pub/linux (Brasil : São Paulo) Slackware  
<ftp.ufpr.br> : /pub/Linux/ (Brasil : Paraná) Slackware

## 16.4 - Lista de discussões (usenet newsgroup)

**comp.os.linux.announce** - é um grupo de anúncios moderado; você deve lê-lo se pretende usar Linux. Submissões a este grupo devem ser mandadas para [linux-announce@news.ornl.gov](mailto:linux-announce@news.ornl.gov).

**comp.os.linux.answers** - Contém todos os FAQs, HOWTOs e outros documentos importantes. Assine este grupo também.

Os outros grupos na hierarquia `comp.os.linux.*` também são recomendados alguns problemas comuns não respondidos neste FAQ podem estar nos newsgroups. Esses grupos são :

**comp.os.linux.setup**  
**comp.os.linux.hardware**  
**comp.os.linux.networking**  
**comp.os.linux.x**  
**comp.os.linux.development.apps,**



`comp.os.linux.development.system`  
`comp.os.linux.advocacy`  
`comp.os.linux.misc.`

