



by Simon Gurney: www.skc.hu

Todos os segredos do Gnome

Pela porta dos fundos

”E o povo já pergunta com maldade: onde estão as opções?” É o que muitos usuários do Gnome podem estar se perguntando, parodiando o antigo sucesso de Noel Rosa. Muitas opções estão escondidas no configurador do Gnome, o GConf. Neste artigo, acendemos a luz da casa das máquinas. **POR ANDREA MÜLLER**

Um tanto caótico e desorganizado, mas altamente configurável – essa era a cantinela que tanto encantou os usuários da antiga série 1.x do Gnome, grupo formado basicamente pelos hackers, nerds e geeks de plantão. A série 2.x mudou completamente essa visão. O projeto Gnome literalmente “parou tudo” e diminuiu a entropia causada pela miríade de opções que confundiam o usuário menos técnico. Daí surgiu o documento *Human Interface Guidelines* (HIG; veja Quadro 1), que criou uma referência para a criação das interfaces gráficas dos programas do Gnome. O resultado de sua aplica-

ção prática em todo o ambiente foi uma melhora significativa na usabilidade do desktop e programas relacionados.

Efeito Colateral

Mesmo que os objetivos sejam louváveis, usuários mais experientes tendem a achar que os desenvolvedores do Gnome foram um pouco longe demais. Se você usa o gerenciador de arquivos Nautilus em outro ambiente que não o Gnome, conhece o velho problema: o Nautilus insiste em recriar o desktop do Gnome dentro do outro ambiente. Antes, os usuários podiam simplesmente desligar a opção *Use Nautilus to draw the*

desktop (Usar o Nautilus para desenhar o desktop). Hoje, se você lançar o Nautilus sem prestar atenção, seu KDE ou WindowMaker será “possuído” pelo Gnome, mesmo que use a opção *--no-desktop* ao iniciar o Nautilus via linha de comando.

O Nautilus também registra a si mesmo como *Help Viewer* (visualizador de ajuda) para os programas do Gnome e uma inocente consulta ao menu *Help | Content* (Ajuda | Conteúdo) em um aplicativo do Gnome que esteja rodando no KDE fará com que o ambiente do usuário seja, novamente, tomado de assalto.

Felizmente, os desenvolvedores não descartaram essas opções: elas simplesmente foram reposicionadas. Em vez dos menus e caixas de diálogo, o GConf (uma espécie de “registro” do Gnome) é o lugar onde todas as opções de configuração do sistema pode ser alteradas. Para manipular o GConf, usamos uma ferramenta chamada, apropriadamente, de *editor do Gconf*.

O Red Hat Linux e o Debian estão entre as poucas distribuições que instalam essa prática ferramenta por padrão. O Mandrake 9.0 e 9.1 possuem a ferramenta, que é instalada como dependência do **metapacote** *gnome2*. Os usuários

Quadro 1: Conhecendo o HIG Human Interface Guidelines

O documento HUMAN INTERFACE GUIDELINES, disponível em <http://developer.gnome.org/projects/gup/hig/>, especifica os aspectos visuais, funcionais e de ergonomia que todos os programas e aplicativos do Gnome deveriam seguir. Os itens abordados incluem disposição consistente dos controles nas janelas, menus bem organizados, traduções e localizações para o maior número possível de idiomas, tudo mantendo o nível de dificuldade mais baixo possível o – por exemplo, retirando do programa as funções que a maioria dos usuários nunca irá usar ou, até mesmo, entender.

O HIG define regras de usabilidade que obrigam os desenvolvedores de software a usar uma linguagem adaptada ao vocabulário do usuário médio e – se a operação sendo executada demorar mais que alguns instantes – providenciar para que o usuário saiba o que o computador está fazendo. Há regras específicas que definem detalhes na disposição dos elementos em uma caixa de diálogo e em menus. É por isso que será muito difícil encontrar um aplicativo do Gnome com caixas de diálogo poluídas, mostrando uma quantidade absurda de botões (por exemplo, *OK, Aplicar, Use este e Fechar* na mesma janela...).

Quando um usuário faz uma mudança numa configuração qualquer, como mostrado na Figura 1, assume-se que quer aplicar a alteração imediatamente. Isso torna redundante o uso de botões separados para *Aplicar* ou *Usar este*. Para que um botão *OK* se os usuários esperam que o botão *Fechar* faça a mesma coisa?

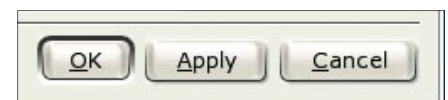


Figura 1: Dois dos três botões nessa caixa de diálogo são redundantes.

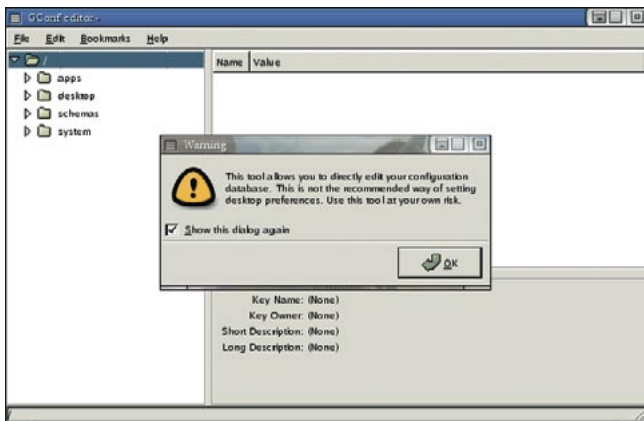


Figura 2: “Use por sua conta e risco!”, é o que ele diz...

do SuSE não têm outra alternativa senão instalar o editor manualmente, mesmo que já tenham o Gnome instalado em seus discos rígidos.

Quase como o Regedit

Ao chamar a ferramenta de configuração com o comando `gconf-editor &`, não deixe o aviso mostrado na figura 2 desencorajá-lo. O programa é, sim, estável e, se você modificar uma opção por engano, sempre é possível revertê-la à configuração padrão.

O programa se parece muito com o editor de registro do Windows®. A estrutura em árvore à esquerda contém as chamadas chaves e subchaves. Os valores são mostrados à direita. Quaisquer mudanças são aplicadas imediatamente e gravadas nos arquivos de configuração do usuário, formatados em XML e armazenados no diretório `~/gconf/`.

Cada chave que contenha um ou mais valores será armazenada num arquivo individual. Por exemplo, `~/gconf/apps/nautilus/preferences/%gconf.xml` contém as configurações atuais do Nautilus, o gerenciador de arquivos do Gnome. Além das configurações individuais dos usuários, há um banco de dados glo-

bal para o sistema, que é usado como base para os usuários que entrem no Gnome pela primeira vez. As definições globais são armazenadas em `/etc/gconf/gconf.xml.defaults`. O SuSE 9 tem seu próprio (fora de padrão) jeito de fazer as coisas, armazenando as configurações globais em `/etc/opt/gnome/gconf/gconf.xml.defaults`. Já o SuSE 8.2 usa o diretório `/etc/opt/gnome2/gconf`. Se você compilou o Gnome a partir dos fontes, o diretório é `/usr/local/etc/gconf`.

Embora o XML seja um formato que usa texto puro, documentos em XML são normalmente difíceis de ler. Como em HTML, cada configuração é marcada pelos sinais de menor (“<”) e maior (“>”) e define o tipo e o valor da opção, entre outras coisas (ver figura 3). O caminho do arquivo XML sempre reflete o caminho usado pelo editor do GConf. Por exemplo, as preferências do Nautilus são armazenadas na chave `/apps/nautilus/preferences`.

O item responsável por desenhar o ambiente de trabalho é chamado `show_desktop`. Ao desligá-lo, o Nautilus é instado a parar de importar uma imagem de fundo do seu KDE. Feche o editor do GConf para aplicar as modificações ao arquivo XML apropriado.

Bem-vindas ao Lar, Opções!

As chaves `apps` e `desktop` são, tipicamente, pontos de partida para localizar configurações escondidas. A chave `apps` é onde se configura o modo como

os aplicativos se comportam. Além dos programas que já vêm com o Gnome, muitos programas externos usam o GConf, como por exemplo o editor de HTML Screem. Já a chave `desktop` é um “sacolé” para configurações globais do próprio ambiente operacional.

É muito difícil relacionar o nome da chave com o que ela realmente faz. As descrições das chaves podem, portanto, ser de grande ajuda. por exemplo, a chave `desktop/gnome/background/picture_filename` especifica a imagem de fundo (ou “papel de parede”) que o Gnome usa no desktop.

Em contraste com a opção “`show_desktop`”, “`picture_filename`” não é um simples caso de ligar e desligar uma característica. Em vez disso, espera-se que o usuário indique o caminho para o arquivo de imagem desejado. Para isso, clique na coluna `Value` de `picture_filename` e digite o caminho completo (ver figura 4). Um duplo clique abre uma caixa de diálogo para editar todos os parâmetros dessa chave (ver figura 5).

Se o nome da opção não é satisfatório, é possível associar a ele uma `docstring`, descrição curta do que a opção faz. Se existir uma `docstring`, o editor do GConf mostrará seu conteúdo no canto inferior direito da tela (ver figura 6).

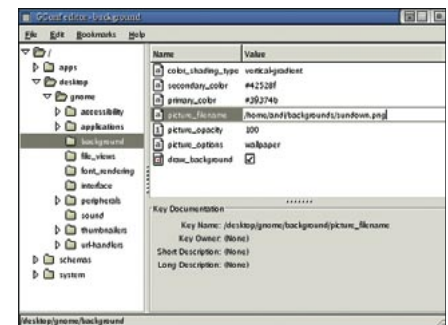


Figura 4: Modificando a imagem de fundo do Gnome no editor do GConf.

GLOSSÁRIO

Metapacote: Um pacote RPM que não contém nenhum arquivo, mas define dependências para outros pacotes. Isso permite instalar um grupo de pacotes, como por exemplo o conjunto de programas que compõem o Gnome, por meio de um único pacote. É um princípio similar ao usado pela ferramenta `tasksel` no Debian.

XML: “Extended Markup Language”, arquivos com sintaxe similar à do HTML mas com uma gama muito maior de funções. É possível não só descrever a estrutura e a formatação de um documento, mas também definir marcações (tags) semânticas, como por exemplo `<City>Londres</City>`. A descrição aplicada ao elemento que está entre as tags pode ser interpretada por programas que entendam esse formato.



Figura 3: O Gnome usa arquivos XML como o mostrado acima para armazenar suas configurações.

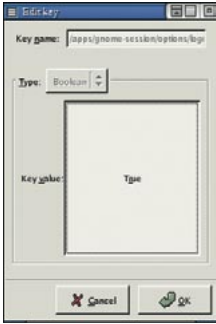


Figura 5: Clique para ligar e desligar a opção.

chaves favoritas pelo menu *Bookmark / Add bookmark*.

Três não é demais

O *gconf-editor* não grava diretamente as opções no banco de dados. Dois programas auxiliares se encarregam disso nos bastidores. Um deles é o comando em modo texto *gconftool-2*, o outro é o servidor de configuração *gconfd*. O editor simplesmente oferece uma maneira gráfica para utilização do *gconftool-2*, que por sua vez repassa as mudanças ao *gconfd*. O *gconfd*, então, repassa as modificações imediatamente para os programas, caso estejam sendo executados. Por exemplo, os ícones, menus e o papel de parede inseridos pelo Nautilus desaparecem imediatamente, sem que seja preciso fechar e abrir novamente esse gerenciador de arquivos.

É possível alterar configurações com o *gconftool-2* diretamente na linha de comando. O comando roda em sistemas em que um front-end gráfico não foi instalado ou que o ambiente gráfico não consiga iniciar. Se suas configurações foram para as cucuia devido a uma falha de energia e o Gnome não consegue iniciar, o *gconftool-2* pode ser seu último recurso antes de uma medida drástica.

Configuração via linha de comando

O *gconftool-2* oferece uma multidão de opções que não são exatamente o que poderíamos chamar de intuitivas. Há uma pequena referência na tabela 2. Alguns exemplos, entretanto, serão mais eficientes para mostrar as possibilidades da ferramenta.

Os comandos normalmente seguem o padrão a seguir:

```
gconftool-2 opção chave
```

É possível navegar pela árvore e ter acesso a opções que não podem ser encontradas nos menus dos aplicativos. A tabela 1 contém uma amostra. É possível, inclusive, guardar suas

Nome da chave no GConf	Efeito
<code>/apps/file-roller/listing/show_mime_icons</code>	Liga/desliga o ícone de tipo de arquivo no File Roller. Quando desabilitado, cada arquivo compactado é mostrado com um ícone branco.
<code>/apps/file-roller/ui/history_len</code>	Número de arquivos no histórico do File Roller
<code>/apps/gedit-2/preferences/editor/save/save_encoding</code>	Especifica a codificação de caracteres usada pelo editor gedit. O padrão é UTF-8. A opção <i>GEDIT_SAVE_ORIGINAL_FILE_ENCODING_IF_POSSIBLE</i> obriga o gedit a salvar arquivos novos em UTF-8 mas deixar intocada a codificação dos já existentes. Isso é muito útil quando se compartilha arquivos com outros usuários e eles reclamam da acentuação.
<code>/apps/gedit-2/preferences/ui/recent/max_recents</code>	Número de arquivos recentes mostrados no menu <i>File</i> (arquivo) do gedit.
<code>/apps/gnome-search-tool/show_additional_options</code>	Mostra opções adicionais de busca (válida no Gnome 2.4 ou posterior).
<code>/apps/nautilus/desktop/home_icon_visible</code>	Especifica se a pasta pessoal do usuário deve ser mostrada na área de trabalho (Gnome 2.4 ou posterior).
<code>/apps/nautilus/desktop/trash_icon_visible</code>	Mostra ou esconde a lixeira (Gnome 2.4 ou posterior).
<code>/apps/panel/global/confirm_panel_remove</code>	Desabilita o pedido de confirmação quando um painel é excluído.
<code>/apps/gnome-session/options/show_splash_screen</code>	Especifica se o Gnome deve mostrar uma tela de abertura, normalmente chamada de <i>splash screen</i> .
<code>/apps/gnome-session/options/splash_image</code>	Caminho para a tela de abertura.
<code>/desktop/gnome/file-views/show_backup_files</code>	Mostra os arquivos de backup, normalmente ocultos, que terminam com um "til" (~). Muitos programas geram cópias de segurança e usam o til para marcá-las. Arquivos iniciados com um ponto continuam ocultos.
<code>/desktop/gnome/file-views/show_hidden_files</code>	Mostra também os arquivos ocultos, i.e., que começam com um ponto.

onde *chave* é o caminho da chave, da mesma forma que seria percorrido pelo editor do GConf, mas sem a última barra. Se precisar alterar o conteúdo de uma chave (em vez de apenas exibi-lo) é necessário informar o valor. Digite as chaves como se fossem caminhos – na verdade, são mesmo. Você pode usar o auto-completar do bash para isso. Um exemplo seria `/apps/nautilus/preferences/show-desktop`. É possível ainda emitir o comando *gconftool-2 --all-dirs /* para mostrar uma lista com todos os caminhos disponíveis:

Glossário

UTF-8: Codificação de caracteres que inclui a maioria dos alfabetos conhecidos pelo homem. Programas com suporte a Unicode são capazes de mostrar caracteres cirílicos, asiáticos e acentuados sem que o usuário tenha que especificar um conjunto diferente de caracteres para um determinado alfabeto. Para mostrar arquivos em Unicode basta que o aplicativo use uma fonte UTF-8.

Splash screen: Uma tela de boas vindas mostrada por um aplicativo sempre que é executado. Normalmente é uma imagem composta por um logotipo e informações sobre autor/fabricante e versão. Além do Gnome, o KDE, o Quanta, o OpenOffice e os drivers da nVidia são exemplos de programas que mostram splash screens.

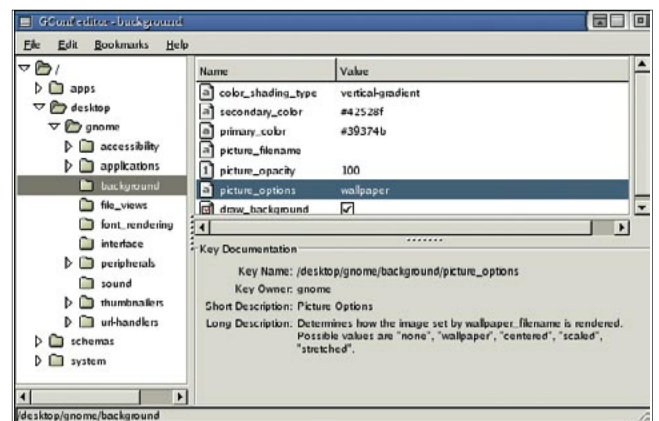


Figura 6: O *docstring* informando uma lista dos possíveis valores para imagens de fundo.

```
/desktop
/apps
/system
/schemas
```

O parâmetro `-a` mostra as opções abaixo de uma determinada chave. `gconftool-2 -a /apps/file-roller/listing` mostra as configurações de tela do compactador *File Roller*. Se precisar exportar as subchaves e opções para um arquivo, a sintaxe seria:

```
gconftool-2 -R /apps >
> apps-options.txt
```

Assim, é possível imprimir o arquivo `apps-options.txt` criado por este exemplo para referência posterior (ver figura 7).

As opções `--long-docs`, `--short-docs` e `-T` mostram detalhes sobre uma opção. As duas primeiras mostram textos de ajuda, se disponíveis. Já `-T` informa o tipo de um determinado valor. Essa é uma informação importante se quiser editar esse valor com o `gconftool-2`.

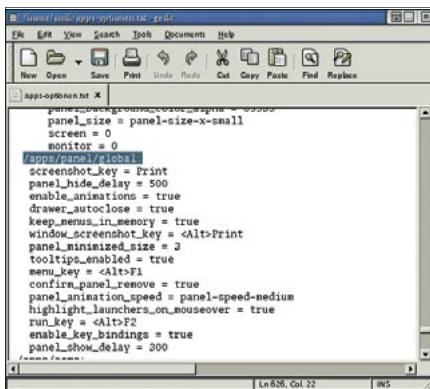


Figura 7: Um arquivo texto com todas as opções de configuração de um aplicativo.

Como regra geral, pode-se assumir que caminhos serão valores do tipo *string* (texto puro), enquanto valores de verdadeiro e falso (liga/desliga) serão do tipo *bool* (álgebra de Boole). Vimos um exemplo disso anteriormente: a opção que especifica se o Nautilus deve ou não meter o dedo na área de trabalho alheia é do tipo *bool*. Há também os tipos *int* (inteiro), *float* (números em ponto flutuante) e *list* (lista). A chave `/apps/panel/default_setup/general/toplevel_id_list` é um exemplo de um tipo lista – neste caso, uma lista que informa os painéis que o Gnome vai carregar quando iniciar.

Me Liga e Conta as Novidades

Depois de descobrir o nome e o tipo de uma chave e de ter certeza do que ela realmente faz, podemos, efetivamente, alterá-la. Por exemplo, o comando a seguir ordena ao Nautilus que pare de bulir com as áreas de trabalho dos outros ambientes:

```
gconftool-2 -t bool -s /apps
/nautilus/preferences
/show_desktop false
```

O parâmetro `-t bool` informa ao `gconftool-2` que tipo de valor está sendo manipulado. Já `-s` indica que se quer alterar uma opção. O valor, indicado no final do comando, é `false`. Especifique `true` ao invés de `false` para reabilitar essa característica. O comando

```
gconftool-2 -t int -s /apps
/nautilus/preferences
/directory_limit 10000
```

aumenta o número máximo de arquivos que o Nautilus pode mostrar. O novo valor é 10000. O Gnome 2.4 (e versões posteriores) não possui restrição; já o Gnome 2.2 restringe esse número para 4000 arquivos. O próximo comando:

```
gconftool-2 -t string /apps
/gnome-session/options
/splash_image home/andi
/mysplash.png
```

ordena ao Gnome que use, no futuro, outra tela de abertura (*splash screen*).

É mais simples ainda alterar uma opção para seu valor padrão, pois não é necessário especificar valor ou tipo. O comando `gconftool-2 -u` seguido da chave apropriada será o suficiente.

Se você fez uma completa lambança nas configurações de um programa, não é necessário repetir o comando para cada opção. O parâmetro `--recursive-unset` restaura todas as opções de um “galho” da árvore para seus respectivos valores padrão. Por exemplo

```
gconftool-2 --recursive-unset
/apps/nautilus
```

restaura a configuração padrão do Nautilus. Isso também funciona para as propriedades da área de trabalho:

```
gconftool-2 --recursive-unset
/desktop
```

Esteja atento, pois isso apagará todas as suas preferências pessoais! Antes de tomar essa atitude drástica, faça um backup do seu diretório `/home/nome_usuario/.gconf` em um lugar bem seguro. Assim, será possível deixar “tudo como dantes no quartel de Abrantes”, caso seja necessário.

Rédea curta

Os parâmetros `--config-source` e `--direct`, com os quais ainda não tivemos a oportunidade de brincar, tornam a vida dos administradores de sistemas extremamente fácil. Eles permitem que se modifique a configuração global do sistema, normalmente presente em `/etc/gconf/gconf.xml.defaults/`. Uma das muitas situações em que isso pode ser aplicado é quando o ambiente gráfico padrão é o KDE e o administrador quer evitar que os usuários menos espertos arruinem sua área de trabalho ao, simplesmente, chamar o Nautilus.

Desative o `gconfd` enquanto modifica as configurações globais, pois esta etapa envolve edição direta dos arquivos XML sem usar o servidor. Encerre qualquer programa do Gnome e use os comandos `top` e `ps -e | grep gconfd` para se certificar de que o servidor não está no ar. Pode levar algum tempo para o servidor ser desativado – seja paciente.

Agora você pode modificar as configurações globais. Por exemplo:

```
gconftool-2 --direct -config
-source xml:readwrite:/etc
/gconf/gconf.xml.defaults/ -t
bool -s /apps/nautilus
/preferences/show_desktop false
```

O parâmetro `xml:readwrite`: força o `gconftool-2` a abrir os arquivos XML em modo de escrita e leitura. Para consultar um valor, digite o comando

```
gconftool-2 --direct -config
-source xml:readonly:/etc/gconf
/gconf.xml.defaults/ -g /apps
/nautilus/preferences
/show_desktop
```

Apenas o superusuário pode rodar esse comando. No SuSE 9, o caminho

Tabela 2: Parâmetros de linha de comando para o gconftool-2

Opção	Argumento	Efeito
-s	chave, opção, valor	Configura o valor da opção; requer o parâmetro -t.
-g	chave, opção	Consulta um valor.
-u	chave, opção	Restaura o Valor padrão.
--recursive-unset	chave	Restaura todos os valores abaixo dessa chave.
-a	chave	Mostra todos os pares opção/valor abaixo dessa chave.
--all-dirs	chave	Mostra as pastas abaixo dessa Chave.
-R	chave	Mostra uma lista de todas as chaves e valores abaixo dessa chave.
-t	tipo	Mostra o tipo de uma determinada chave-- <i>int, string, bool, float, list</i> ou <i>pair</i> .
-T	chave, opção	Mostra o tipo de uma determinada opção.
--short-docs, --long-docs	chave, opção	Mostra a documentação detalhada para uma opção.
--config-source	banco de dados de configuração	Especifica para qual conjunto de configurações (ou seja, de qual usuário) a operação se aplica. Usuários comuns têm permissão de leitura, apenas, para as configurações globais.
--direct		Modifica o banco de dados sem usar o servidor. Isso é necessário para mudar qualquer configuração que não seja a do próprio usuário atualmente usando o sistema, ou seja, qualquer configuração que não esteja em seu próprio <i>~/.gconf</i> . Desative o servidor <i>gconfd</i> antes de fazer esse tipo de alteração.

para o banco de dados de configuração é */etc/opt/gnome/gconf/gconf.xml.defaults*. Usuários do SuSE 8.2 devem substituir *gnome* por *gnome2*.

Usuários comuns podem fazer as mesmas alterações, mas apenas em seus próprios bancos de dados, que estão em sua pasta pessoal. Para impedir que seus usuários mudem as próprias configurações – e evitar chamados desnecessários ao *help desk* – é possível bloquear certos aspectos do desktop. Isso envolve um terceiro banco de dados, o */etc/gconf/gconf.xml.mandatory/*, que armazena valores obrigatórios que os usuários não podem modificar.

Se o Gnome é seu ambiente padrão e você quer impedir que seus usuários mudem as configurações do Nautilus, o comando a seguir é um ótimo exemplo de como resolver o problema:

```
gconftool-2 --direct -config
-source xml:readwrite:/etc
/gconf/gconf.xml.mandatory
/ -t bool -s /apps/nautilus
/preferences/show_desktop true
```

O editor do GConf ainda mostrará a opção, mas os usuários não poderão alterá-la. No Gnome 2.2, um bug fez com que sucessivas ações do mouse numa mesma opção travassem o programa.

Nos casos em que as opções estejam disponíveis tanto no GConf quanto nos menus do aplicativo que está sendo

configurado, as restrições só serão respeitadas se o usuário ainda não tiver mexido nas configurações. Para evitar isso, restaure os padrões com o parâmetro *-u*. Depois é possível – por exemplo – remover a função que mostra o comando apagar “verdadeiro” no Nautilus. O comando para tal é:

```
gconftool-2 --direct -config
-source xml:readwrite:/etc
/gconf/gconf.xml.mandatory
/ -t bool -s /apps/nautilus
/preferences/enable_delete false
```

Os usuários que tentarem habilitar essa opção serão agraciados com uma mensagem de erro (ver figura 8).

Se esse aperitivo atiçou seu apetite, a refeição completa será de lamber os beiços. A documentação de referência e um mundo de exemplos podem ser encontrados na ajuda do Gnome, em *Área de Trabalho / System Administration Guide / Using Gconf* (a documentação ainda não está traduzida para o português). ■

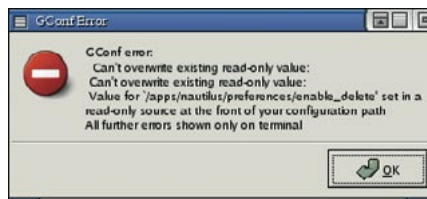


Figura 8: Os usuários não têm permissão de apagar arquivos diretamente, mas ainda podem enviá-los à lixeira.

Qual é a do GConf?

O principal motivo para o desenvolvimento do GConf é tornar a configuração de aplicativos mais fácil para os administradores de sistema, centralizando-a em um só local. O GConf é “transparente”, o que significa que, se uma configuração for modificada em um aplicativo, é atualizada automaticamente em todos os outros aplicativos por ela afetados. Esta tecnologia é vital para o recurso “aplicar instantaneamente” do Gnome 2 e fundamental na escrita de aplicativos formados por múltiplos componentes.

O GConf também é expansível. Sua API foi criada pensando em facilitar futuras modificações; o “back-end” que armazena os dados de configuração pode ser substituído sem maiores problemas. Além do back-end XML atual, o GConf também pode usar, para armazenar as configurações, arquivos binários (como no registro do Windows), um servidor de catálogo LDAP ou mesmo um banco de dados SQL.

E, ao contrário do registro do Windows, cada chave pode ter documentação anexa explicando seu propósito. Além disso, um mecanismo de travas impede que múltiplos aplicativos tentem modificar uma chave simultaneamente, evitando o temível problema da corrupção dos dados.

INFORMAÇÕES

- [1] *Gnome*: <http://www.gnome.org>
- [2] *Página oficial do GConf*: <http://www.gnome.org/projects/gconf/>
- [3] *Gnome User Interface Guidelines 2.0*: <http://developer.gnome.org/projects/gup/hig/>
- [4] *Apple User Interface Guidelines*: <http://developer.apple.com/documentation/UserExperience/Conceptual/OSXHI-Guidelines/index.html>
- [5] *Introduction to the GConf Library*: <http://developer.gnome.org/feature/archive/gconf/gconf.html>

SOBRE A AUTORA

Após quase dois anos como jornalista independente, Andrea Mueller agora trabalha como editora para a Linux New Media AG.



Quando não está lidando com artigos ou empacotando software para a área de downloads da revista *EasyLinux*, ela gosta de ir além do Linux e se envolve com outros sistemas operacionais, como o QNX, o BeOS e o NetBSD.