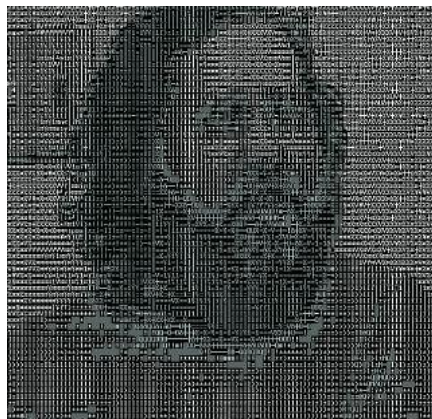


## Ferramentas gráficas do ImageMagick

# Gráficos em modo texto



Para a maioria das pessoas, manipulação de imagens é sinônimo de uma interface gráfica complexa, com inúmeras paletas cheias de ferramentas, seleções de cor e coisas do gênero. Infelizmente, existem muitas tarefas para as quais um programa como o Gimp é simplesmente inapropriado.

O ImageMagick [1], que muitas distribuições Linux instalam automaticamente, tem uma seleção de ferramentas em linha de comando para a manipulação de imagens. Trataremos de algumas delas nesse artigo.

## Quadro 1: Script “color2gray”

```
01 #!/bin/bash
02 # Criando arquivos TIF e PDF
   monocromáticos
03 #
04 base=$(echo $1 | sed -e
   "s/\....$/") # Remove a
   extensão do arquivo
05 baseBW=${base}.tif
   # Nome do arquivo TIF
06 basePDF=${base}.pdf
   # Nome do arquivo PDF
07 convert -colors 256 -
   colorspace \
08 GRAY $1 $baseBW
   # Converte em TIF
09 convert $baseBW $basePDF
   # ... e em PDF
10 mkdir -p COLOR; mv $1 COLOR/
   # Altera original
```

O Gimp é o que a maioria das pessoas escolhe como aplicativo padrão para tarefas mais complexas de manipulação de imagem. Contudo, para muitas tarefas, como redimensionamento e conversão de formatos, a linha de comando oferece um caminho mais rápido. **POR HANS-GEORG ESSER**

## Mudando Formatos

Converter imagens para outro formato, como TIF em PNG, é uma tarefa bastante comum. O comando do ImageMagick que faz isso é *convert*:

```
convert image.tif image.png
```

O *convert* avalia a extensão do arquivo em questão para definir o formato final. Não faz nenhuma diferença digitar a extensão em minúsculas ou maiúsculas, ou mesmo uma mistura das duas. A ferramenta lida até mesmo com variantes como *.tiff* ao invés vez de *.tif*, ou *.jpeg* ao invés de *.jpg*. Você pode digitar:

```
man ImageMagick
```

para abrir a página principal do manual (manpage) do ImageMagick e obter uma lista dos formatos de arquivo suportados.

## Scripting

Uma das principais vantagens das ferramentas de linha de comando é o fato de poderem ser usadas em um script para automatizar as tarefas repetitivas.

Vamos imaginar que você precise converter regularmente screenshots a cores no formato PNG em arquivos TIF monocromáticos e em arquivos PDF em escala de cinza. Para fazê-lo, simplesmente use o script *color2gray* mostrado no Quadro 1. Ele não apenas converte o formato

PNG em TIF e PDF, como também reduz o universo de cores. As opções que manipulam isso são *-colors 256* e *-color space GRAY*.

O script também se encarrega de parte da organização do processo. Se você digitar o comando abaixo:

```
color2gray test.jpg
```

ele cria *test.tif* e *test.pdf* e move o arquivo original para um subdiretório chamado *COLOR* – se esse diretório não existir, o script simplesmente o cria.

## Redimensionando

Na editora, precisamos ter gráficos de diferentes tamanhos da página de capa para cada edição (ver Figura 2).

O *convert* nos ajuda a fazer isso, usando a opção *-geometry* para mudar o tamanho da imagem. O tamanho final é especificado como *largura x altura*, como no comando:

```
convert big.png -geometry 150x200 small.png
```

para uma imagem pequena com uma largura até 150 pixels e uma altura até 200 pixels. Se você quer que o *convert* utilize exatamente esses valores (e distorça a imagem ao fazê-lo), tecle um sinal de exclamação imediatamente depois dos dados de tamanho:

## Quadro 2: Script “conversão de documento”

```
01 #!/bin/sh
02 echo converting title: $1 to $2 \ (small\ ) and $3 \ (very small\ )
03 convert $1 -blur 3 -geometry 220x312! $2
04 convert $1 -blur 3 -geometry 149x211! $3
```

```
convert big.png -geometry 2
150x200! small.png
```

Aqui na editora, usamos a listagem mostrada no Quadro 2 para preparar os gráficos da capa que serão publicados na Web. No nosso caso, faz sentido usar um tamanho fixo (ou seja, o parâmetro '!'); pois isso nos assegura que todas as miniaturas sejam do mesmo tamanho, além de nos permitir criar um layout com as miniaturas alinhadas uma ao lado da outra.

Além da conversão manual de arquivos, o *convert* também tem uma propriedade especial que cria um panorama com todas as imagens. Digite os seguintes comandos em um diretório contendo apenas imagens:

```
convert 'vid:*' directory.png
```

para criar um panorama com as miniaturas das imagens e os nomes dos arquivos, resolução e tamanho como legenda sobre fundo branco.

Você pode usar imagens desse tipo como “mapas de imagens clicáveis” em sites na Web. O utilitário *convert* pode ser usado para criar automaticamente destes mapas. O comando a seguir irá criar três arquivos:

```
convert 'vid:*' HTML:index.html
```

- *index.gif*: é a imagem em miniatura. O *convert* deduz o nome da imagem a partir do nome do arquivo HTML.

### Quadro 3: Comando de Identificação

```
01 $ identify *.png *.jpg
02 john@linux:~/LMUK45/Cover> identify *.png *.jpg
03 ald15.png PNG 1704x1710+0+0 PseudoClass 256c 8-bit 877.4kb 0.150u 2
0:01
04 Cover_45.png[1] PNG 620x879+0+0 DirectClass 8-bit 345.0kb 0.040u 0:01
05 Cover_45s.png[2] PNG 220x312+0+0 DirectClass 8-bit 69.4kb 0.010u 0:01
06 Cover_45ss.png[3] PNG 149x211+0+0 DirectClass 8-bit 36.8kb 0.000u 2
0:01
07 directory.png[4] PNG 768x186+0+0 DirectClass 16-bit 268.6kb 0.020u 2
0:01
08 Manpage.png[5] PNG 636x966+0+0 DirectClass 8-bit 48.8kb 0.030u 0:01
09 little_duck.jpg[6] JPEG 800x600+0+0 DirectClass 8-bit 59.5kb 0.030u 2
0:01
```

## Entrada e Saída

O *import* e o *display* são ferramentas bastante úteis. O primeiro cria screenshots, o outro mostra imagens. O *import* tem uma vasta gama de opções. Para capturar toda a área de trabalho, digite:

```
import -window root 2
desktop.png
```

E para capturar uma janela:

```
import -frame window.png
```

O parâmetro *-frame* é obrigatório, uma vez que, sem ele, o *import* remove a moldura

da janela e captura apenas o conteúdo.

Se você usa o ambiente de trabalho KDE, pode pressionar *[Alt + F2]* para abrir uma linha de comando, e experimentar usar o comando *import* em conjunto com o comando *sleep*:

```
sleep 3; import -window root 2
desktop.png
```

com isso você ganha algum tempo para organizar sua área de trabalho antes que o comando seja executado. Você pode, por exemplo, abrir alguns menus.

Finalmente, o *display* é um visualizador simples que mostra as imagens em seu tamanho real. Se não há espaço suficiente para isso, a ferramenta mostra uma janela de pré-visualização, permitindo que o usuário selecione as partes da imagem que lhe interessam.

## Conclusão

*convert*, *identify*, *import* e *display* não são mais que uma pequena seleção do kit de ferramentas do ImageMagick e discutimos apenas superficialmente suas características. Para distribuições baseadas em RPM ou Debian, use os seguintes comandos para descobrir que programas estão incluídos no pacote:

```
rpm -qfl `which identify` | 2
grep bin
dpkg -L ImageMagick | grep bin
```

E lembre-se: quando tudo mais falhar, leia as páginas do manual! ■

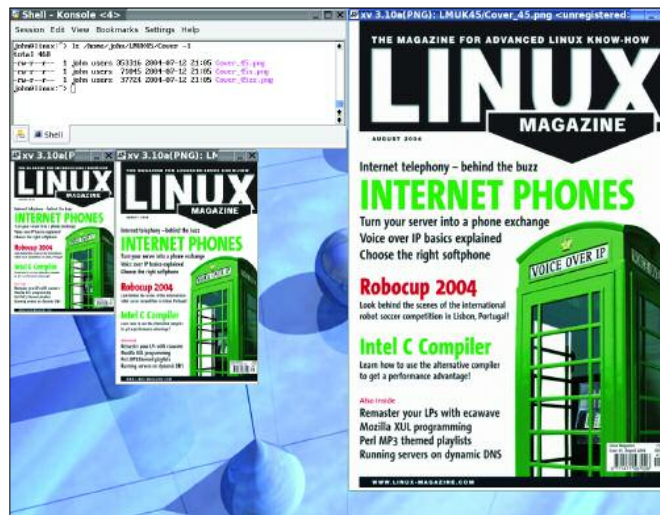


Figura 1: Usamos versões de diferentes tamanhos da imagem de capa na página da Linux Magazine.

- *index.html*: é um arquivo HTML com a imagem GIF embutida e devidamente mapeada; basta dar um clique na imagem em miniatura ser levado à imagem maior.
- *index\_map.shtml*: contém a seção de mapeamento de *index.html* e é útil para integração (no servidor) com múltiplos sites na Web.

## Detalhes

O comando *identify* também faz parte do pacote ImageMagick. Ele exibe informações sobre os arquivos que passam por ele, incluindo o tamanho em pixels (veja o Quadro 3) e a profundidade de cor (8 ou 16 bit no nosso exemplo).

O *identify* não utiliza a extensão do arquivo para identificar seu formato - mesmo se ela for removida, o utilitário continuará reconhecendo o arquivo. Ele indica o tamanho da imagem no formato *largura x altura*, assim como o *convert*.