

# Fontes e FreeBSD

## Resumo

Este documento contém a descrição de vários arquivos de fontes que podem ser usados no FreeBSD e no driver do console do sistema, X11, Ghostscript e Groff. Manuais passo-a-passo são fornecidos para alterar a exibição do console do sistema para o modo 80x60, e para usar fontes tipo 1 com os aplicativos acima.

---

## Índice

1. Introdução .....	1
2. Terminologia Básica .....	1
3. Quais Formatos de Fonte eu Posso Utilizar? .....	2
4. Configurando um Console Virtual para o Modo de Linhas 80x60 .....	3
5. Usando Fontes Type 1 com X11 .....	3
6. Usando Fontes Type 1 com Ghostscript .....	7
7. Usando Fontes Type 1 com Groff .....	7
8. Convertendo Fontes TrueType para um Formato groff/PostScript Para o Groff .....	10
9. As Fontes TrueType Podem ser Usadas com Outros Programas? .....	13
10. Onde Fontes Adicionais Podem ser Obtidas? .....	13
11. Questões Adicionais .....	13

## 1. Introdução

Existem muitas origens de fontes disponíveis, e alguém pode questionar como elas podem ser utilizadas no FreeBSD. A resposta pode ser encontrada numa busca cuidadosa na documentação do componente onde você gostaria de utilizar a mesma. Isto pode consumir um bom tempo, portanto esse tutorial é uma tentativa de fornecer um atalho para outros que possam estar interessados.

## 2. Terminologia Básica

Existem muitos formatos diferentes de fontes e sufixos de arquivos de fontes associados. Alguns deles serão abordados aqui:

### **.pfa, .pfb**

PostScript® fonte tipo 1. O .pfa é o formato Ascii e o .pfb é o formato Binário .

### **.afm**

As métricas da fonte associado com a fonte tipo 1.

---

### **.pfm**

As métricas da fonte para impressora associadas com a fonte tipo 1.

### **.ttf**

Uma fonte TrueType®

### **.fot**

Uma referência indireta para uma fonte TrueType (não é realmente uma fonte)

### **.fon, .fnt**

Fontes de tela bitmap

O arquivo .fot é usado pelo Windows® como um tipo de link simbólico para o arquivo de fonte TrueType® (.ttf). Os arquivos de fonte .fon também são utilizados no Windows. Eu desconheço uma maneira de utilizar essa fonte no FreeBSD.

## **3. Quais Formatos de Fonte eu Posso Utilizar?**

Qual formato de arquivo de fonte é útil depende do aplicativo que está sendo usado. O FreeBSD por si só, não usa fontes. Aplicativos e/ou drivers podem fazer uso de arquivos de fontes. Aqui tem uma pequena referência cruzada de aplicação/driver para os sufixos de tipo de fonte:

### **Driver**

**vt**

.hex

**syscons**

.fnt

### **Aplicativo**

**Ghostscript**

.pfa, .pfb, .ttf

**X11**

.pfa, .pfb

**Groff**

.pfa, .afm

**Povray**

.ttf

O sufixo .fnt é frequentemente utilizado. Suspeito que, sempre que alguém quisesse criar um arquivo de fonte especializado para seu aplicativo, na maioria das vezes eles escolhiam esse sufixo. Então, é provável que os arquivos com esse sufixo não tenham o mesmo formato; Especificamente,

os arquivos .fnt usados pelo console do sistema do FreeBSD pode não ser do mesmo formato que o arquivo .fnt encontrado no ambiente MS-DOS®/Windows®. Não fiz nenhuma tentativa de utilizar outro arquivo .fnt senão aquele fornecido com o FreeBSD.

## 4. Configurando um Console Virtual para o Modo de Linhas 80x60

Primeiro, uma fonte 8x8 deve ser carregada. Para fazer isso, /etc/rc.conf deve conter a linha (altere o nome da fonte para uma apropriada para a sua região):

```
font8x8="iso-8x8"      # font 8x8 from /usr/shared/syscons/fonts/* (or NO).
```

O comando para alterar o modo é [vidcontrol\(1\)](#):

```
% vidcontrol VGA_80x60
```

Vários programas orientados à tela, como o [vi\(1\)](#), devem ser capazes de determinar a dimensão corrente da tela. Como isto é conseguido através de uma chamada do [ioctl](#) para o driver do console (tal como o [syscons\(4\)](#)) ele irá determinar corretamente as dimensões da nova tela.

Para fazer isso de uma maneira mais integrada, é possível incluir esses comandos nos scripts de inicialização de modo que ocorra quando o sistema for iniciado. Para fazer isso basta adicionar essa linha no /etc/rc.conf

```
allscreens_flags="VGA_80x60"  # Set this vidcontrol mode for all virtual screens
```

Referências: [rc.conf\(5\)](#), [vidcontrol\(1\)](#).

## 5. Usando Fontes Type 1 com X11

O X11 pode tanto usar o formato .pfa quanto o formato .pfb de fonte. As fontes do X11 estão localizadas em vários subdiretórios abaixo do /usr/X11R6/lib/X11/fonts. Cada arquivo de fonte é uma referência cruzada do seu nome X11 com o conteúdo dos arquivos fonts.dir em cada diretório.

Já existe um diretório chamado Type1. A forma mais direta de adicionar uma nova fonte é colocá-la nesse diretório. Uma forma melhor seria colocar todas as novas fontes num diretório separado e usar um link simbólico para as fontes adicionais. Isso permite identificar as fontes sem confundir com aquelas que são originalmente fornecidas. Por exemplo:



```

% strings showboat.pfb | more
%!FontType1-1.0: Showboat 001.001
%%CreationDate: 1/15/91 5:16:03 PM
%%VMusage: 1024 45747
% Generated by Fontographer 3.1
% Showboat
  1991 by David Rakowski.  Alle Rechte Vorbehalten.
FontDirectory/Showboat known{/Showboat findfont dup/UniqueID known{dup
/UniqueID get 4962377 eq exch/FontType get 1 eq and}{pop false}ifelse
{save true}{false}ifelse}{false}ifelse
12 dict begin
/FontInfo 9 dict dup begin
 /version (001.001) readonly def
 /FullName (Showboat) readonly def
 /FamilyName (Showboat) readonly def
 /Weight (Medium) readonly def
 /ItalicAngle 0 def
 /isFixedPitch false def
 /UnderlinePosition -106 def
 /UnderlineThickness 16 def
 /Notice (Showboat
 1991 by David Rakowski.  Alle Rechte Vorbehalten.) readonly def
end readonly def
/FontName /Showboat def
--stdin--

```

Usando essas informações, um possível nome poderia ser:

```
-type1-Showboat-medium-r-normal-decorative-0-0-0-0-p-0-iso8859-1
```

Os componentes do nosso nome são:

### Companhia

Vamos nomear todas as novas fontes como `type1`.

### Família

O nome da fonte

### Densidade

Normal, negrito, média, semi-negrito, etc. Pelas informações acima do `strings(1)`, essa fonte aparenta ter uma densidade *média*.

### Inclinação

roman, itálico, oblíquo, etc. Como o *Ângulo Itálico* é zero, o *roman* será utilizado.

### Largura

Normal, ampla, condensada, estendida, etc. Até que possa ser examinada, supomos que será

*normal*.

## Estilo Adicional

Frequentemente omitido, mas isso indicará que a fonte possui letras maiúsculas decorativas.

## Espaçamento

proporcional ou monoespaçada. A opção *Proporcional* é usada quando o *isFixedPitch* é falso.

Todos esses nomes são arbitrários, mas deve-se tentar ser compatível com as convenções existentes. A fonte é referenciada pelo nome com possíveis curingas pelo programa X11, então o nome escolhido tem que fazer algum sentido. Pode-se começar simplesmente usando

```
···-normal-r-normal-···-p-···
```

como o nome, e então usar [xfontsel\(1\)](#) para examiná-lo e ajustar o nome com base na aparência da fonte.

Então, para completar nosso exemplo:

```
Torne a fonte acessível para o X11
% cd /usr/X11R6/lib/X11/fonts/Type1
% ln -s /usr/local/shared/fonts/type1/showboat.pfb .

Edite os arquivos fonts.dir e fonts.scale, adicionando a linha que descreve a fonte e
incremente o número de fontes que são encontradas na primeira linha.
% ex fonts.dir
:1p
25
:1c
26
.
:$a
showboat.pfb -type1-showboat-medium-r-normal-decorative-0-0-0-0-p-0-iso8859-1
.
:wq

fonts.scale aparenta ser idêntico ao fonts.dir...
% cp fonts.dir fonts.scale

Informe ao X11 que arquivos foram alterados
% xset fp rehash

Examine a nova fonte
% xfontsel -pattern -type1-*
```

Referências: [xfontsel\(1\)](#), [xset\(1\)](#), The X Windows System in a Nutshell, [O'Reilly & Associates](#).

## 6. Usando Fontes Type 1 com Ghostscript

O Ghostscript referencia uma fonte via seu arquivo de Fontmap. Este deve ser modificado de uma maneira similar a feita para o arquivo fonts.dir do X11. O Ghostscript pode usar tanto o formato .pfa quanto o .pfb. Usando a fonte do exemplo anterior, segue um passo a passo de como utilizá-la com o Ghostscript:

```
Coloque a fonte no diretório do Ghostscript
% cd /usr/local/shared/ghostscript/fonts
% ln -s /usr/local/shared/fonts/type1/showboat.pfb .

Edite o mapeamento de fontes, assim o Ghostscript saberá sobre a fonte
% cd /usr/local/shared/ghostscript/4.01
% ex Fontmap
:$a
/Showboat      (showboat.pfb) ; % From CICA /fonts/atm/showboat
.
:wq

Use o Ghostscript para checar a fonte
% gs prfont.ps
Aladdin Ghostscript 4.01 (1996-7-10)
Copyright (C) 1996 Aladdin Enterprises, Menlo Park, CA. All rights
reserved.
This software comes with NO WARRANTY: see the file PUBLIC for details.
Loading Times-Roman font from /usr/local/shared/ghostscript/fonts/tir____.pfb...
 /1899520 581354 1300084 13826 0 done.
GS>Showboat DoFont
Loading Showboat font from /usr/local/shared/ghostscript/fonts/showboat.pfb...
 1939688 565415 1300084 16901 0 done.
>>showpage, press <return> to continue<<
>>showpage, press <return> to continue<<
>>showpage, press <return> to continue<<
GS>quit
```

Referências: Veja o arquivo fonts.txt na distribuição do Ghostscript 4.01

## 7. Usando Fontes Type 1 com Groff

Agora que a nova fonte pode ser utilizada tanto pelo X11 quanto pelo Ghostscript, como podemos utilizar a nova fonte com o Groff? Primeiro de tudo, como estamos usando fontes Type 1 PostScript®, o dispositivo Groff que é aplicável é o device *ps*. Um arquivo de fonte deve ser criado para cada fonte que o Groff possa usar. Um nome de fonte Groff é apenas um arquivo no `/usr/shared/groff_font/devps`. Com o nosso exemplo, o arquivo da fonte poderia ser `/usr/shared/groff_font/devps/SHOWBOAT`. O arquivo deve ser criado usando ferramentas providas pelo Groff.

A primeira ferramenta é o `afmtodit`. Ela normalmente não está instalada, então deve ser baixada de uma fonte de distribuição. Eu percebi que teria que mudar a primeira linha do arquivo, então eu fiz:

```
% cp /usr/src/gnu/usr.bin/groff/afmtodit/afmtodit.pl /tmp
% ex /tmp/afmtodit.pl
:1c
#!/usr/bin/perl -P-
.
:wq
```

Essa ferramenta irá criar o arquivo de fontes do Groff a partir dos arquivos de métrica (.afm suffix.) Continuando com nosso exemplo:

```
Muitos arquivos .afm estão no formato do Mac, com ^M delimitando as linhas
Nós temos que convertê-los para o estilo UNIX que delimita as linhas com ^J
% cd /tmp
% cat /usr/local/shared/fonts/type1/showboat.afm |
    tr '\015' '\012' >showboat.afm

Agora crie um arquivo de fonte groff
% cd /usr/shared/groff_font/devps
% /tmp/afmtodit.pl -d DESC -e text.enc /tmp/showboat.afm generate/textmap SHOWBOAT
```

A fonte agora pode ser referenciada pelo nome SHOWBOAT.

Se o Ghostscript é utilizado para impressão com driver simulado no sistema, então não precisa fazer mais nada. Entretanto, se as impressoras usam PostScript® real, então a fonte deve ser baixada pela impressora de maneira que a fonte a ser utilizada (a menos que a impressora tenha a fonte showboat embutida ou acessível a partir de um disco de fontes.) O passo final é criar uma fonte descarregável. A ferramenta `pfbtops` é usada para criar o formato .pfa da fonte, e o arquivo para download é modificado para referenciar a nova fonte. O arquivo para download deve referenciar o nome interno da fonte. Isso pode ser facilmente determinado de um arquivo de fonte groff conforme demonstrado:

```
Criando o arquivo de fonte .pfa

% pfbtops /usr/local/shared/fonts/type1/showboat.pfb >showboat.pfa
```

Claro que, se o arquivo .pfa já existe, apenas crie um link simbólico para referenciá-lo.



Obtendo o nome interno da fonte

```
% fgrep internalname SHOWBOAT
```

```
internalname Showboat
```

Dizendo ao groff que tem que fazer o download da fonte

```
% ex download
```

```
:$a
```

```
Showboat      showboat.pfa
```

```
.
```

```
:wq
```

Para testar a fonte:

```

% cd /tmp

% cat >example.t <<EOF
.sp 5
.ps 16
This is an example of the Showboat font:
.br
.ps 48
.vs (\n(.s+2)p
.sp
.ft SHOWBOAT
ABCDEFGHI
.br
JKLMNOPQR
.br
STUVWXYZ
.sp
.ps 16
.vs (\n(.s+2)p
.fp 5 SHOWBOAT
.ft R
To use it for the first letter of a paragraph, it will look like:
.sp 50p
\s(48\f5H\s0\fRere is the first sentence of a paragraph that uses the
showboat font as its first letter.
Additional vertical space must be used to allow room for the larger
letter.
EOF
% groff -Tps example.t >example.ps

Para usar ghostscript/ghostview
% ghostview example.ps

Para imprimí-la
% lpr -Ppostscript example.ps

```

Referências: [/usr/src/gnu/usr.bin/groff/afmtodit/afmtodit.man](#), [groff\\_font\(5\)](#), [groff\\_char\(7\)](#), [pfbtops\(1\)](#).

## 8. Convertendo Fontes TrueType para um Formato groff/PostScript Para o Groff

Isso potencialmente requer um pouco de trabalho, simplesmente porque depende de alguns utilitários que não são instalados como parte do sistema base. Eles são:

### **ttf2pf**

Utilitário de conversão TrueType para PostScript. Ele permite a conversão de uma fonte

TrueType em um arquivo de métrica de fonte ascii (.afm).

Atualmente disponível em <http://sunsite.icm.edu.pl/pub/GUST/contrib/BachoTeX98/ttf2pf/>. Nota: Esses arquivos são programas PostScript e devem ser baixados para o disco mantendo pressionada a tecla **Shift** ao clicar no link. Caso contrário, seu navegador pode tentar iniciar o ghostview para visualizá-los.

Os arquivos de interesse são:

- GS\_TTF.PS
- PF2AFM.PS
- ttf2pf.ps

O caso engraçado sobre maiúsculas/minúsculas é devido ao fato de serem destinados também para os terminais DOS. O ttf2pf.ps faz uso dos outros como maiúsculos, portanto, qualquer renomeação deve ser consistente com isso. (Na verdade, GS\_TTF.PS e PFS2AFM.PS são supostamente parte da distribuição Ghostscript, mas é muito fácil usá-los como utilitários isolados. O FreeBSD parece não incluir o último.) Você também pode querer instalá-los em `/usr/local/shared/groff_font/devps` (?).

### afmtodit

Cria arquivos de fontes para uso com o Groff a partir do arquivo de métricas de fonte ascii. Isso geralmente fica no diretório `/usr/src/contrib/groff/afmtodit` e requer algum trabalho para prosseguir.



Se você é paranóico sobre o trabalhar no diretório `/usr/src`, simplesmente copie o conteúdo do diretório acima para um local de trabalho.

Na área de trabalho, você precisará compilar o utilitário. Apenas digite:

```
# make -f Makefile.sub afmtodit
```

Você também pode precisar copiar o `/usr/contrib/groff/devps/generate/textmap` para `/usr/shared/groff_font/devps/generate` se ele ainda não existir.

Depois que todos esses utilitários estiverem no lugar, você está pronto para começar:

1. Crie o arquivo .afm digitando:

```
% gs -dNODISPLAY -q -- ttf2pf.ps TTF_name PS_font_name AFM_name
```

Onde, *TTF\_name* é o seu arquivo de fonte TrueType, *PS\_font\_name* é o nome do arquivo .pfa, *AFM\_name* é o nome que você deseja para o arquivo .afm. Se você não especificar nomes de arquivos de saída para os arquivos .pfa ou .afm, os nomes padrão serão gerados a partir do nome do arquivo de fonte TrueType.

Isso também produz um arquivo .pfa, o arquivo ascii de métricas de fonte PostScript (.pfb é para o formato binário). Isso não será necessário, mas poderia (eu acho) ser útil para um fontserver.

Por exemplo, para converter a fonte de código de barras 3of9 usando o nome de arquivo padrão, use o seguinte comando:

```
% gs -dNODISPLAY -- ttf2pf.ps 3of9.ttf
Aladdin Ghostscript 5.10 (1997-11-23)
Copyright (C) 1997 Aladdin Enterprises, Menlo Park, CA. All rights reserved.
This software comes with NO WARRANTY: see the file PUBLIC for details.
Converting 3of9.ttf to 3of9.pfa and 3of9.afm.
```

Se você quiser que as fontes convertidas sejam armazenadas em A.pfa e B.afm, use este comando:

```
% gs -dNODISPLAY -- ttf2pf.ps 3of9.ttf A B
Aladdin Ghostscript 5.10 (1997-11-23)
Copyright (C) 1997 Aladdin Enterprises, Menlo Park, CA. All rights reserved.
This software comes with NO WARRANTY: see the file PUBLIC for details.
Converting 3of9.ttf to A.pfa and B.afm.
```

## 2. Crie o arquivo PostScript Groff:

Vá para o diretório `/usr/shared/groff_font/devps` para facilitar a execução do comando abaixo. Você provavelmente precisará de privilégios de root para isso. (Ou, se você é paranoico sobre trabalhar lá, certifique-se de referenciar os arquivos DESC, text.enc e generate/textmap como estando neste diretório.)

```
% afmtodit -d DESC -e text.enc file.afm generate/textmap PS_font_name
```

Onde, `file.afm` é o `AFM_name` criado pelo `ttf2pf.ps` acima e `PS_font_name` é o nome da fonte usada a partir desse comando, bem como o nome que `groff(1)` usará para referências a essa fonte. Por exemplo, supondo que você usou o primeiro comando `ttf2pf.ps` acima, a fonte 3of9 Barcode pode ser criada usando o comando:

```
% afmtodit -d DESC -e text.enc 3of9.afm generate/textmap 3of9
```

Assegure-se de que o arquivo `PS_font_name` resultante (por exemplo, 3of9 no exemplo acima) esteja localizado no diretório `/usr/shared/groff_font/devps` copiando-o ou movendo-o para lá.

Note que se o `ttf2pf.ps` atribuir um nome de fonte usando o nome que ele encontrou no arquivo de fonte TrueType e você quiser usar um nome diferente, você deverá editar o arquivo .afm antes de executar o `afmtodit`. Esse nome também deve coincidir com o usado no arquivo Fontmap se você deseja redirecionar o `groff(1)` para o `gs(1)`.

## 9. As Fontes TrueType Podem ser Usadas com Outros Programas?

O formato de fonte TrueType é usado pelo Windows, Windows 95 e Mac. É bastante popular e há um grande número de fontes disponíveis neste formato.

Infelizmente, há poucos aplicativos que conheço que podem usar este formato: O Ghostscript e o Povray são os que vem a mente. O suporte do Ghostscript, de acordo com a documentação, é rudimentar e os resultados provavelmente serão inferiores as fontes Type 1. O Povray versão 3 também tem a capacidade de usar fontes TrueType, mas eu duvido que muitas pessoas criem documentos como uma série de páginas tridimensionais traçadas com luz :-).

Esta situação bastante triste pode mudar em breve. O [Projeto FreeType](#) está atualmente desenvolvendo um conjunto útil de ferramentas FreeType:

- O servidor de fontes `xfsft` para X11 pode fornecer fontes TrueType além de fontes regulares. Embora esteja atualmente em beta, dizem que está bastante utilizável. Veja [a página de Juliusz Chroboczek](#) para maiores informações. Instruções de portabilidade para o FreeBSD podem ser encontradas na [página do software de Stephen Montgomery](#).
- O `xfstt` é outro servidor de fontes para o X11, disponível em <ftp://sunsite.unc.edu/pub/Linux/X11/fonts/>.
- Um programa chamado `tft2bdf` pode produzir arquivos BDF adequados para uso em um ambiente X a partir de arquivos TrueType. Os binários para o Linux estão disponíveis em <ftp://crl.nmsu.edu/CLR/multiling/Geral/>.
- e outros ...

## 10. Onde Fontes Adicionais Podem ser Obtidas?

Muitas fontes estão disponíveis na Internet. Elas são totalmente gratuitas ou sharewares. Além disso, muitas fontes estão disponíveis na categoria `x11-fonts/` na coleção do ports

## 11. Questões Adicionais

- Quais são os usos dos arquivos `.pfm`?
- Posso gerar o arquivo `.afm` a partir de um arquivo `.pfa` ou `.pfb`?
- Como gerar os arquivos de mapeamento de caracteres `groff` para fontes PostScript com nomes de caracteres não padrão?
- Podem os dispositivos `xditview` e `devX` serem configurados para acessar todas as novas fontes?
- Seria bom ter exemplos de uso de fontes TrueType com Povray e Ghostscript.