

# Fontes e FreeBSD

## Resumo

Este documento contém uma descrição dos vários arquivos de fontes que podem ser usados com o FreeBSD e o driver syscons, X11, Ghostscript e Groff. Exemplos de receitas são fornecidos para alternar o display syscons para o modo 80x60 e para usar fontes type 1 com os programas de aplicativos mencionados 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 o X11 .....	3
6. Usando Fontes Type 1 com Ghostscript .....	6
7. Usando Fontes Type 1 com o Groff .....	7
8. Convertendo Fontes TrueType para um Formato groff/PostScript para o Groff .....	9
9. As Fontes TrueType Podem ser Usadas com Outros Programas? .....	11
10. Onde é possível obter fontes adicionais? .....	12
11. Questões Adicionais .....	12

## 1. Introdução

Existem muitas fontes disponíveis e alguém pode perguntar como usá-las com o FreeBSD. A resposta pode ser encontrada pesquisando cuidadosamente a documentação do componente que deseja utilizar. Isso pode ser muito demorado, portanto, este tutorial é uma tentativa de fornecer um atalho para outras pessoas interessadas.

## 2. Terminologia Básica

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

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

Fontes PostScript® type 1. O arquivo .pfa é a forma Ascii e o arquivo .pfb é a forma Binária.

### **.afm**

As métricas da fonte associado com a fonte type 1.

---

### **.pfm**

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

### **.ttf**

Uma fonte TrueType®

### **.fot**

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

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

Fontes de tela bitmapped

O arquivo .fot é usado pelo Windows® como um tipo de link simbólico para o arquivo de fonte TrueType® real (.ttf). Os arquivos de fonte .fon também são usados pelo Windows. Não conheço uma maneira de usar esse formato de fonte no FreeBSD.

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

O formato de arquivo de fonte mais adequado depende do aplicativo utilizado. O FreeBSD por si só não utiliza fontes. Programas de aplicativos e/ou drivers podem fazer uso dos arquivos de fonte. Aqui está uma pequena referência cruzada de aplicativos/drivers para os sufixos de tipo de fonte:

### **Driver**

**vt**

.hex

**syscons**

.fnt

### **Aplicação**

**Ghostscript**

.pfa, .pfb, .ttf

**X11**

.pfa, .pfb

**Groff**

.pfa, .afm

**Povray**

.ttf

O sufixo .fnt é bastante utilizado. Eu suspeito que sempre que alguém quisesse criar um arquivo de fonte especializado para seu aplicativo, na maioria das vezes escolhia esse sufixo. Portanto, é provável que os arquivos com esse sufixo não sejam todos do mesmo formato; especificamente, os

arquivos .fnt usados pelo syscons no FreeBSD podem não ser do mesmo formato que um .fnt encontrado no ambiente MS-DOS®/Windows®. Não fiz nenhuma tentativa de usar outros arquivos .fnt além daqueles fornecidos com o FreeBSD.

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

Primeiramente, uma fonte 8x8 deve ser carregada. Para isso, o arquivo `/etc/rc.conf` deve conter a linha (mude o nome da fonte para um apropriado para sua localização):

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

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

```
% vidcontrol VGA_80x60
```

Vários programas orientados a tela, como [vi\(1\)](#), devem ser capazes de determinar as dimensões atuais da tela. Como isso é alcançado por meio de chamadas `ioctl` para o driver do console (como [syscons\(4\)](#)), eles determinarão corretamente as novas dimensões da tela.

Para tornar isso mais contínuo, é possível incorporar esses comandos nos scripts de inicialização para que ocorram durante a inicialização do sistema. Para fazer isso, adicione esta linha ao arquivo `/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 o X11

O X11 pode usar fontes em formato .pfa ou .pfb. As fontes X11 estão localizadas em vários subdiretórios em `/usr/X11R6/lib/X11/fonts`. Cada arquivo de fonte é cruzado com seu nome X11 pelos conteúdos de `fonts.dir` em cada diretório.

Já existe um diretório chamado `Type1`. A maneira mais direta de adicionar uma nova fonte é colocá-la neste diretório. Uma maneira melhor é manter todas as novas fontes em um diretório separado e usar um link simbólico para a fonte adicional. Isso permite que você mantenha um controle mais fácil das suas fontes sem confundir com as fontes fornecidas originalmente. Por exemplo:

```
Crie um diretório para conter os arquivos de fonte
% mkdir -p /usr/local/share/fonts/type1
% cd /usr/local/share/fonts/type1
```

Coloque aqui os arquivos .pfa, .pfb e .afm

Pode ser desejável manter arquivos readme e outras documentações

para as fontes aqui também

```
% cp /cdrom/fonts/atm/showboat/showboat.pfb .
% cp /cdrom/fonts/atm/showboat/showboat.afm .
```

Mantenha um índice para cruzar as fontes de referência.

```
% echo showboat - InfoMagic CICA, Dec 1994, /fonts/atm/showboat >>INDEX
```

Agora, para usar uma nova fonte com o X11, é necessário tornar o arquivo de fonte disponível e atualizar os arquivos de nome de fonte. Os nomes das fontes X11 se parecem com:

```
-bitstream-charter-medium-r-normal-xxx-0-0-0-0-p-0-iso8859-1
|         |         |         |         |         |         |         |         |         |         |         \         \
|         |         |         |         |         |         |         |         |         |         |         +-----+ character set
|         |         |         |         |         |         |         |         |         |         |         \         \ +- average width
|         |         |         |         |         |         |         |         |         |         |         |         \         \ +- spacing
|         |         |         |         |         |         |         |         |         |         |         |         \         \ +- vertical res.
|         |         |         |         |         |         |         |         |         |         |         |         \         \ +- horizontal res.
|         |         |         |         |         |         |         |         |         |         |         |         \         \ +- points
|         |         |         |         |         |         |         |         |         |         |         |         |         \         \ +- pixels
|         |         |         |         |         |         |         |         |         |         |         |         |         \         \
foundry  family  weight  slant  width  additional style
```

Um novo nome precisa ser criado para cada nova fonte. Se você tiver alguma informação da documentação que acompanha a fonte, ela pode servir como base para criar o nome. Se não houver informações disponíveis, você pode ter uma ideia usando o comando `strings(1)` no arquivo de fonte. 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 apenas nomear todas as novas fontes como `type1`.

### Família

O nome da fonte.

### Densidade

Normal, negrito (bold), médio (medium), seminegrito (semibold), etc. Pelos resultados do comando `strings(1)` mostrados acima, parece que esta fonte tem um peso *médio* (medium).

### Inclinação

romano (roman), itálico (italic), oblíquo (oblique), etc. Já que o ângulo de itálico (*ItalicAngle*) é zero, será usada a versão romana (*roman*).

### Largura

Normal, ampla (wide), condensada (condensed), estendida (extended), etc. Até que possa ser examinada, a suposição será de que a fonte é *normal*.

### Estilo Adicional

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

### Espaçamento

Proporcional (proportional) ou espaçamento fixo (monospaced). Será usada a versão *proporcional*, já que o valor de *isFixedPitch* é falso.

Todos esses nomes são arbitrários, mas deve-se procurar ser compatível com as convenções existentes. Uma fonte é referenciada pelo nome, com possíveis curingas (wildcards) por um programa X11, então o nome escolhido deve fazer algum sentido. Pode-se começar simplesmente usando

```
...-normal-r-normal-...-p-...
```

esse nome, e depois usar o comando `xfontsel(1)` para examiná-lo e ajustar o nome com base na aparência da fonte.

Então, para completar nosso exemplo:

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

Edite o arquivo fonts.dir e fonts.scale, adicionando a linha que descreve a fonte
e incrementando o número de fontes que é encontrado 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

O arquivo fonts.scale parece ser idêntico ao fonts.dir...
% cp fonts.dir fonts.scale

Informe ao X11 que as coisas mudaram.
% 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 por meio de seu arquivo Fontmap. Este arquivo deve ser modificado de maneira semelhante ao arquivo fonts.dir do X11. O Ghostscript pode usar fontes em formato .pfa ou .pfb. Usando a fonte do exemplo anterior, aqui está como usá-la com o Ghostscript:

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

Edite o arquivo Fontmap para que o Ghostscript saiba sobre a fonte
% cd /usr/local/share/ghostscript/4.01
```

```

% ex Fontmap
:$a
/Showboat      (showboat.pfb) ; % From CICA /fonts/atm/showboat
.
:wq

Use o Ghostscript para examinar 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/share/ghostscript/fonts/tir____.pfb...
/1899520 581354 1300084 13826 0 done.
GS>Showboat DoFont
Loading Showboat font from /usr/local/share/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: arquivo fonts.txt na distribuição do Ghostscript 4.01

## 7. Usando Fontes Type 1 com o Groff

Agora que a nova fonte pode ser usada tanto pelo X11 quanto pelo Ghostscript, como se pode usá-la com o groff? Em primeiro lugar, uma vez que estamos lidando com fontes PostScript® type 1, o dispositivo groff que é aplicável é o dispositivo *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 em `/usr/share/groff_font/devps`. Em nosso exemplo, o arquivo de fonte poderia ser `/usr/share/groff_font/devps/SHOWBOAT`. O arquivo deve ser criado usando ferramentas fornecidas pelo groff.

A primeira ferramenta é `afmtodit`. Ela não costuma ser instalada por padrão, então é necessário obtê-la da distribuição de origem. Descobri que precisei alterar a primeira linha do arquivo, então fiz o seguinte:

```

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

```

Essa ferramenta criará o arquivo de fonte do groff a partir do arquivo de métricas (sufixo `.afm`). Continuando com nosso exemplo:

Muitos arquivos .afm estão no formato Mac, com linhas delimitadas por ^M.  
É necessário convertê-los para o estilo UNIX(R), com linhas delimitadas por ^J.

```
% cd /tmp
% cat /usr/local/share/fonts/type1/showboat.afm |
  tr '\015' '\012' >showboat.afm
```

Agora crie o arquivo de fonte do groff.

```
% cd /usr/share/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 for usado para controlar as impressoras no sistema, então nada mais precisa ser feito. No entanto, se forem usadas impressoras PostScript® reais, a fonte deve ser baixada para a impressora para que a fonte seja usada (a menos que a impressora tenha a fonte showboat incorporada ou em um disco de fonte acessível). O último passo é criar uma fonte transferível. A ferramenta `pfbtops` é usada para criar o formato .pfa da fonte, e o arquivo download é modificado para referenciar a nova fonte. O arquivo download deve fazer referência ao nome interno da fonte. Isso pode ser facilmente determinado a partir do arquivo de fonte do groff, como ilustrado:

Crie o arquivo de fonte .pfa.

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

Claro, se o arquivo .pfa já estiver disponível, basta usar um link simbólico para fazer referência a ele.

```
Obtenha o nome interno da fonte
% fgrep internalname SHOWBOAT
internalname Showboat
Informe ao groff que a fonte deve ser baixada.
% ex download
:$a
Showboat      showboat.pfa
.
:wq
```

Para testar a fonte:

```
% cd /tmp
% cat >exemplo.t <<EOF
.sp 5
.ps 16
Este é um exemplo da fonte Showboat:
.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
Para usá-lo como a primeira letra de um parágrafo, ficará assim:
.sp 50p
\s(48\f5H\s0\fRqui está a primeira frase de um parágrafo que usa a
fonte showboat como a primeira letra.
Deve-se usar espaço vertical adicional para permitir espaço para a letra maior.
EOF
% groff -Tps example.t > exemplo.ps

Para usar o ghostscript/ghostview
% ghostview example.ps

Para imprimir:
% 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 pode requerer um pouco de trabalho, simplesmente porque depende de algumas ferramentas que não são instaladas como parte do sistema base. Elas são:

### ttf2pf

Utilitários de conversão de TrueType para PostScript. Isso permite a conversão de uma fonte TrueType para um arquivo de métricas de fonte ascii (.afm).

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

Os arquivos de interesse são:

- GS\_TTF.PS
- PF2AFM.PS

- ttf2pf.ps

A utilização de letras maiúsculas e minúsculas engraçadas deve-se ao fato de serem destinadas também para shells do DOS. O arquivo ttf2pf.ps faz uso dos outros em letras maiúsculas, então qualquer renomeação deve ser consistente com isso. (Na verdade, os arquivos GS\_TTF.PS e PFS2AFM.PS supostamente fazem parte da distribuição do Ghostscript, mas é fácil usá-los como uma fonte isolada. O FreeBSD não parece incluir o último.) Você também pode querer tê-los instalados em /usr/local/share/groff\_font/devps(?).

### afmtodit

A ferramenta `afmtodit` é utilizada para criar arquivos de fonte para o groff a partir de um arquivo métrico de fonte em formato ASCII (.afm). Normalmente, essa ferramenta está localizada no diretório /usr/src/contrib/groff/afmtodit, mas normalmente exige alguma configuração para funcionar corretamente.



Se você está preocupado com a segurança de trabalhar no diretório /usr/src, basta copiar o conteúdo do diretório acima para uma localização de trabalho.

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

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

Você também pode precisar copiar o arquivo /usr/contrib/groff/devps/generate/textmap para /usr/share/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\_nome* é o nome do arquivo da sua fonte TrueType, *nome\_fonte\_PS* é o nome do arquivo para .pfa, *nome\_AFM* é o nome desejado para .afm. Se você não especificar os nomes de arquivo de saída para os arquivos .pfa ou .afm, então nomes padrão serão gerados a partir do nome do arquivo da fonte TrueType.

Isso também produz um arquivo .pfa, o arquivo de métricas de fonte PostScript em formato ASCII (.pfb é para a forma binária). Isso não será necessário, mas poderia (eu acredito) ser útil para um servidor de fontes.

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 quer que as fontes convertidas sejam armazenadas em A.pfa e B.afm, use o seguinte 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 groff PostScript:

Acesse o diretório `/usr/share/groff_font/devps` para facilitar a execução do seguinte comando. É provável que você precise de privilégios de root para executá-lo. (Ou, se você é paranoico em relação a trabalhar lá, certifique-se de que os arquivos `DESC`, `text.enc` e `generate/textmap` estejam localizados 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 usado no comando anterior, bem como o nome que `groff(1)` usará para referenciar essa fonte. Por exemplo, assumindo que você usou o primeiro `ttf2pf.ps` acima, a fonte Barcode 3of9 pode ser criada usando o comando:

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

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

Observe que, se o `ttf2pf.ps` atribuir um nome de fonte usando o que ele encontra no arquivo de fonte TrueType e você desejar usar um nome diferente, será necessário editar o arquivo `.afm` antes de executar o `afmtodit`. Este nome também deve corresponder ao usado no arquivo `Fontmap` se você quiser redirecionar o `groff(1)` para o `gs(1)`.

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

O formato de fonte TrueType é utilizado pelo Windows, Windows 95 e Mac's. É bastante popular e há uma grande quantidade de fontes disponíveis neste formato.

Infelizmente, há poucos aplicativos que eu conheço que podem usar este formato: o Ghostscript e o Povray vêm à mente. O suporte do Ghostscript, de acordo com a documentação, é rudimentar e os resultados provavelmente serão inferiores aos das fontes type 1. A versão 3 do Povray também tem

a capacidade de usar fontes TrueType, mas eu duvido que muitas pessoas estejam criando documentos como uma série de páginas renderizadas em raytracing :-).

Essa situação um tanto quanto desanimadora pode mudar em breve. O [Projeto FreeType](#) está atualmente desenvolvendo um conjunto de ferramentas FreeType úteis:

- O servidor de fontes `xfstt` para X11 pode servir fontes TrueType além de fontes regulares. Embora esteja atualmente em beta, é dito que ele é bastante utilizável. Veja a página de [Juliusz Chroboczek](#) para obter mais informações. As instruções de portabilidade para FreeBSD podem ser encontradas em [página de software de Stephen Montgomery](#).
- `xfstt` é outro servidor de fontes para X11, disponível em <ftp://sunsite.unc.edu/pub/Linux/X11/fonts/>.
- Um programa chamado `ttf2bdf` pode produzir arquivos BDF adequados para uso em um ambiente X a partir de arquivos TrueType. Binários para Linux estão disponíveis em <ftp://crl.nmsu.edu/CLR/multiling/General/>.
- e outros...

## 10. Onde é possível obter fontes adicionais?

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

## 11. Questões Adicionais

- Para que servem os arquivos `.pfm`?
- É possível gerar o arquivo `.afm` a partir de um arquivo `.pfa` ou `.pfb`?
- Como gerar os arquivos de mapeamento de caracteres do `groff` para fontes PostScript com nomes de caracteres não-padrão?
- É possível configurar o `xditview` e os dispositivos `devX` para acessar todas as novas fontes?
- Seria bom ter exemplos de como usar fontes TrueType com o `Povray` e o `Ghostscript`.