フォントと FreeBSD

概要

ここでは FreeBSD の syscons ドライバや X11, Ghostscript, Groff で利用することができるさまざまなフォントファイルについて説明しています。 また、syscons ディスプレイを 80x60 行モードに切り替える方法や、 上述のアプリケーションでタイプ 1 フォントを利用する方法を例示します。

目次

1. はじめに . . . 1
2. 基本用語 . . . 1
3. どのフォント形式を利用できますか? . . . 2
4. 仮想コンソールを 80x60 行モードに設定する . . . 3
5. タイプ 1 フォントを X11 で利用する . . . 3
6. タイプ 1 フォントを Ghostscript で利用する . . . 6
7. タイプ 1 フォントを Groff で利用する . . . 7
8. TrueType フォントを groff 用に groff/PostScript フォーマットに変換する . . . 9
9. TrueType フォントを他のプログラムで使うことができますか? . . . 11
10. どこでフォントを入手できますか? . . . 12
11. その他の質問 . . . 12

1. はじめに

数多くのフォントのソースを入手することができますが、これらを FreeBSD でどのようにして使うかはあまりよく知られていないかもしれません。
その答えは、使いたいと思う構成要素の説明書を注意深く探すことによって見つけることができます。
しかし、これはとても時間がかかる作業です。本チュートリアルは、
フォントに関して興味がある向きに、 その近道を教えようと試みるものであります。

2. 基本用語

フォント形式の種類やそれに関連したフォントファイルの拡張子は多数存在します。
その内でここで解説するものは以下の通りです。

.pfa、.pfb

PostScript® タイプ 1 フォント。拡張子 .pfa は *A*scii 形式のそして拡張子 .pfb は *B*inary 形式を意味する。

.afm

タイプ 1 フォントに関連するフォントメトリック情報。

.pfm

タイプ 1 フォントに関連するプリンタ用フォントメトリック情報。

.ttf

TrueType® フォント。

.fot

TrueType フォントへの間接的な参照ファイル (実際にはフォントファイルではない)。

.fon、.fnt

スクリーン表示用ビットマップフォント。

.fot ファイルは、Windows® で用いられ、実際の TrueType® フォント (.ttf) ファイルへのシンボリックリンクに類する役割を果たします。.fon フォントも Windows で用いられていますが、 FreeBSD でこの形式のフォントを利用する方法を筆者は知りません。

3. どのフォント形式を利用できますか?

どのフォントファイル形式が有用であるかは、利用するアプリケーションに依ります。FreeBSD 自身はフォントファイルは利用しません。 アプリケーションプログラムやドライバ (あるいはその両方) によっては、あるフォントファイルを利用するようにできるかもしれません。
以下は、アプリケーション、及び、
ドライバとそれが利用できるフォントタイプの拡張子の対応表を簡単に示します。

ドライバ

syscons

.fnt

アプリケーション

Ghostscript

.pfa、 .pfb、 .ttf

X11

.pfa、 .pfb

Groff

.pfa、 .afm

Povray

.ttf

拡張子 .fnt は極めて頻繁に使われています。(訳注: この拡張子がフォント (font) という名前から連想しやすいので)
あるアプリケーションに特化したフォントを作成しようとした際にはいつでも、
この拡張子が選択される方がそうでないときよりもかなり多いのではないかと著者は疑っています。
このため、この拡張子を持つファイル全てが同じ形式にはなっていないようです。 特に、.fnt ファイルは FreeBSD 上では syscons によって利用されていますが、これと MS-DOS® や Windows® 環境で出会った .fnt とは同じ形式ではないかもしれません。筆者は FreeBSD で提供されている以外の .fnt

ファイルを利用する試みは一切行っていません。

4. 仮想コンソールを 80x60 行モードに設定する

まず、8x8 サイズのフォントがロードされていなくてはなりません。 そのためには、/etc/rc.conf に以下の行が含まれているべきです (フォントの名称をあなたの locale に対応するものに書き換えてください)。

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

実際にモードを切り替えるコマンドは vidcontrol(1) です。

```
% vidcontrol VGA_80x60
```

vi(1) のような、さまざまなスクリーン指向のプログラムに対して、現在の画面サイズが分かるようにしておかなくてはなりません。これは `ioctl` を通じて (syscons(4) などの) コンソールドライバに呼び掛けることで行われ、

これらを一度に済ませるために、 これらのコマンドを起動用のスクリプトに書いておき、これをシステム起動時に実行するかもしれません。 この方法では /etc/rc.conf に以下の行を追加します

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

参考文献: rc.conf(5)、vidcontrol(1)

5. タイプ 1 フォントを X11 で利用する

X11 では、 .pfa 形式、もしくは、 .pfb 形式のフォントのどちらでも利用できます。 X11 では、フォントは /usr/X11R6/lib/X11/fonts 以下のさまざまなサブディレクトリに置かれています。それぞれのディレクトリにある fonts.dir ファイルの内容によって、 それぞれのフォントのファイルと X11 上でのフォント名が関連付けられています。

Type1 という名前のディレクトリが既に存在しています。 新しいフォントを追加する最も簡単な方法は、このディレクトリのそのフォントファイルを置くことです。
新しいフォントは別なディレクトリに置いておき、Type1
ディレクトリに追加フォントへのシンボリックリンクを張る方がより優れています。
なぜなら、この方法をとることでオリジナルで供給されているフォントと混乱することなく、
これらのフォントを追加した跡を残すことがより簡単にできるからです。
この方法は、例えば、次のように行います。

```
フォントファイルを入れるディレクトリを作成します。
% mkdir -p /usr/local/shared/fonts/type1
% cd /usr/local/shared/fonts/type1

ここに .pfa または .pfb ファイルと .afm ファイルを置きます。
```

```
フォントの readme ファイルやその他のドキュメントをこのディ

レクトリに置いても構いません。
% cp /cdrom/fonts/atm/showboat/showboat.pfb .
% cp /cdrom/fonts/atm/showboat/showboat.afm .

フォントのクロスリファレンスのためにインデックスを変更します。
% echo showboat - InfoMagic CICA, Dec 1994, /fonts/atm/showboat >>INDEX
```

さて、新しいフォントを X11 で利用するためには、 そのフォントファイルを利用できるようにして、フォント名のファイルを更新する必要があります。X11 のフォント名は次のようになっています。

```
-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
```

新しいフォントそれぞれに対して、新しい名前を付ける必要があります。
フォント付属のドキュメントにフォントに関する情報があれば、
名前を作る際の基になるかもしれません。そのような情報がない場合は、 フォントに対して strings(1) を使うと何らかのアイデアが得ることができます。例えば、

```
% 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--
```

この情報から、次のような名前が考えられます:

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

この名前の構成は次の通りです。

型 **(foundry)**

新フォントは type1 と名付けることにしましょう。

族 **(family)**

フォントの名前です。

重み **(weight)**

normal (普通)、bold (太い)、medium (中間)、 semibold (やや太め) などがあります。上記の strings(1) の出力より、 フォントの重みは *medium* であると考えられます。

傾斜 **(slant)**

*r*oman (ローマン体)、*i*talic (イタリック体)、*o*blique (斜字体) などがあります。 *ItalicAngle* が 0になっていることにより、 *roman* を使っています。

幅

normal (普通)、wide (幅広)、condensed (圧縮)、extended(拡張) などがあります。上記で調べた結果から、 *normal* を仮定します。

追加スタイル

通常は省略されますが、フォントに装飾用 (decorative) 英大文字が含まれていることをここで示します。

スペーシング

proportional (プロポーショナル (訳注: 字形に応じて幅が変化するフォント)) または monospaced (単一幅フォント) があります。ここでは *Proportional* としてありますが、これは *isFixedPitch* が false (偽) になっているためです。

これらの名前は全て任意なのですが、 既存の慣習と互換性を保つよう努力すべきでしょう。 X11 プログラムでは、 フォントはワイルドカードを含んだ名前で参照されます。ですから、 フォント名は何らかの意味づけを持って選択されるべきでしょう。 (訳注 : 適当なフォントを探すとき、) ある人は単純に以下の名前を使うことから始めるかもしれません。

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

そして、xfontsel(1)　　で該当するフォントを調べてみて、そのフォントの形を見ながら、名前を調節するかもしれません。

それでは、ここまでの例を完結させることにしましょう。

```
X11 に対してフォントをアクセスできるようにします。
% cd /usr/X11R6/lib/X11/fonts/Type1
% ln -s /usr/local/shared/fonts/type1/showboat.pfb .

fonts.dir と fonts.scale
を編集して、フォントを記述する行を追加し、最初の行にある総フォント数を増やします。
% 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 は fonts.dirと同一内容のようですので...
% cp fonts.dir fonts.scale

X11 に内容が変更されたことを伝えます。
% xset fp rehash

新しいフォントを試してみます。
% xfontsel -pattern -type1-*
```

参考文献: xfontsel(1)、xset(1)、The X Windows System in a Nutshell、O'Reilly & Associates

6. タイプ 1 フォントを Ghostscript で利用する

Ghostscript では、 Fontmap ファイルに従ってフォントを参照して います。このファイルを X11 の fonts.dir ファイルと同様な方法で変更しなくてはなりません。 Ghostscript では、 .pfa 形式または .pfb 形式のフォントのいずれか一方を使用することができます。
前章の例で登場したフォントを使って、ここではこのフォントを　　Ghostscript で使用する方法について述べます。

```
フォントを Ghostscript のフォントディレクトリに置きます。
% cd /usr/local/shared/ghostscript/fonts
% ln -s /usr/local/shared/fonts/type1/showboat.pfb .

Ghostscript にフォントを認識させるために Fontmap を編集します。
% cd /usr/local/shared/ghostscript/4.01
% ex Fontmap
:$a
```

```
/Showboat        (showboat.pfb) ; % From CICA /fonts/atm/showboat
.
:wq

Ghostscript を用いてフォントを試してみます。
% 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
```

参考文献: Ghostscript バージョン 4.01 で配布されている fonts.txt

7. タイプ 1 フォントを Groff で利用する

ここまでで新しいフォントを X11 と Ghostscript の両方で用いることができるようになりましたが、この新しいフォントをどのようにすれば Groff で使うことができるでしょうか? まず第一に、PostScript® のタイプ 1 フォントを扱っていますから、これを適用できる Groff デバイスは *ps* デバイスです。次に、各々のフォントを Groff で使用できるようにフォントファイルを作らなくてはなりません。Groff でのフォント名は /usr/shared/groff_font/devps の中のファイル名になります。上述の例では、フォントファイルは /usr/shared/groff_font/devps/SHOWBOAT とすることができるでしょう。このファイルは Groff によって提供されているツールを用いて生成しなくてはなりません。

最初に `afmtodit` というツールを使います。このコマンドは通常ではインストールされませんので、ソースプログラム群から該当プログラムを取り出さなくてはなりません。
このファイルの最初の一行を変更しなくてはならないことが分かっています。
著者は次のようにしました。

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

このツールはメトリックファイル (.afm 拡張子) から Groff フォントファイルを生成してくれます。フォント使用方法例を続けることにしましょう。

```
.afm ファイルの多くは Mac 形式... すなわち行が ^M で区切られています。
これを行を ^J で区切る UNIX(R) スタイルに変換する必要があります。
% cd /tmp
% cat /usr/local/shared/fonts/type1/showboat.afm |
    tr '\015' '\012' >showboat.afm

そして、groff フォントファイルを生成します。
% cd /usr/shared/groff_font/devps
% /tmp/afmtodit.pl -d DESC -e text.enc /tmp/showboat.afm generate/textmap SHOWBOAT
```

これでフォントを SHOWBOAT という名前で参照することができました。

システムでプリンタを扱うために Ghostscript を使用しているならば、これで作業は完了しました。しかしながら、本物の PostScript® プリンタを使っている場合は、フォントを使用可能にする為に、当該フォントをプリンタにダウンロードする必要があります (showboat フォントがプリンタに偶然にも最初から組み込まれている場合、もしくはプリンタからアクセスされるフォントディスクの中に入っている場合はこの限りではありません)。フォント利用の最終段階として、ダウンロード可能な形式のフォントを生成します。 ツール pfbtops は (訳注 : .pfb 形式から) .pfa 形式のフォントを生成するために、そして、 download というファイルを編集し、フォントの内部名を参照するように変更しなくてはなりません。 この内部名は以下で示すように groff フォントファイルから容易に調べることができます。

```
.pfa フォントファイルを生成する。
% pfbtops /usr/local/shared/fonts/type1/showboat.pfb >showboat.pfa
```

もちろん、.pfa が既に利用可能であれば、 参照できるようにシンボリックリンクを張って下さい。

```
内部フォント名を得る。
% fgrep internalname SHOWBOAT
internalname Showboat
該当フォントをダウンロードしなくてはならないことを groff に通知する。
% ex download
:$a
Showboat      showboat.pfa
.
:wq
```

フォントを試用する。

```
% 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

ghostscript/ghostviewを使って表示する。
% ghostview example.ps

印刷する（訳注 : プリンタ名は適宜変更して下さい)。
% lpr -Ppostscript example.ps
```

参考文献: /usr/src/gnu/usr.bin/groff/afmtodit/afmtodit.man、groff_font(5)、groff_char(7)、pfbtops(1)

8. TrueType フォントを groff 用に groff/PostScript フォーマットに変換する

これにはいくつかユーティリティが必要ですが、
ベースシステムの一部としてインストールされてはいないので若干の作業が必要となります。
インストールするものは:

`ttf2pf`

TrueType から PostScript への変換ユーティリティです。これは TrueType フォントからアスキーフォントメトリック (.afm) ファイルへの変換を行います。

現時点では http://sunsite.icm.edu.pl/pub/GUST/contrib/BachoTeX98/ttf2pf/ から入手できます。
注意: これらのファイルは PostScript によるプログラムなので、`Shift` キーを押しながらリンクをクリックして ディスクにダウンロードしてください。
さもないとあなたのブラウザは ghostview を立ちあげます。

重要なファイルは:

- GS_TTF.PS
- PF2AFM.PS
- ttf2pf.ps

大文字と小文字が奇妙に混在しているのは、 DOS シェルのことも考慮しているためです。 ttf2pf.ps はそれ以外のファイルを 大文字として扱いますので、 ファイル名の変更はそれに対応させてください (実際には GS_TTF.PS と PFS2AFM.PS は、一応 Ghostscript の配布物の一部ですが、個別のユーティリティとしても問題なく利用できます。 FreeBSD には後者が入っていないようです)。 /usr/local/shared/groff_font/devps にインストールするとよいかもしれません。

afmtodit

はアスキーフォントメトリックファイルから Groff とともに使うフォントファイルを作ります。これは通常、 /usr/src/contrib/groff/afmtodit ディレクトリに存在していて、使えるようにするには作業が必要です。



もしも /usr/src ツリーで作業をすることを躊躇うなら、このディレクトリの内容を作業用の場所にコピーすればいいです。

作業エリアで以下のようにしてこのユーティリティします。

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

もし、まだ存在していなければ /usr/contrib/groff/devps/generate/textmap を /usr/shared/groff_font/devps/generate にコピーします。

これらのユーティリティが所定の場所に収まったら いつでも開始できます。

1. .afm ファイルを以下のようにして作ります。

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

ここで、*TTF_name* はあなたの TrueType フォントの名前で、*PS_font_name* は .pfa ファイルのためのファイル名で、 *AFM_name* は .afm ファイルに望む名前です. .pfa や .afm 用の出力ファイル名を明示しなければ、 デフォルト名は TrueType フォントファイル名から作成されます。

この時、アスキー PostScript フォントメトリックファイルである .pfa ファイルも同時に作られます (.pfb はバイナリ形式です)。 これは不要となるでしょうが、(私が考えるに) フォントサーバには役立つでしょう。

例として、30f9 バーコードフォントをデフォルトのファイル名で変換するには以下のようにします。

```
% 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.
```

変換後のフォントを A.pfa と B.afm にするなら以下のようにします。

```
% 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. Groff PostScript ファイルを作ります。

   以下のコマンドの実行が用意なように /usr/shared/groff_font/devps に ディレクトリを変更します。恐らく root 特権が必要になるでしょう (そこでの作業が気にいらないなら、このディレクトリの DESC、 text.enc、 generate/textmap ファイルが参照されるということに注意してください)。

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

   ここで、file.afm は *AFM_name* で、上で ttf2pf.ps で作ったものです。 *PS_font_name* はコマンドから使われるフォント名で、 groff(1) がこのフォントを参照するために使うものです。たとえば、最初の tiff2pf.ps コマンドを上述のように行っていたとすると、 3of9 バーコードフォントは以下のコマンドで作成できます。

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

   得られる *PS_font_name* ファイル (この例では 3of9) はディレクトリ /usr/shared/groff_font/devps に、コピーするなり移動するなりして置かれることに気をつけてください。

   ttf2pf.ps がわりつけるフォント名は TrueType フォントファイル中に見つかったものになります。それとは異なる名前を使いたかったら、 .afm ファイルを編集してから afmtodit を実行する必要があります。 groff(1) から gs(1) へパイプするつもりならば、その名前は同時にフォントマップファイルで使われているものである必要があります。

9. TrueType フォントを他のプログラムで使うことができますか?

TrueType フォント形式は Windows、Windows 95、Mac で用いられます。この形式は極めて有名であり、 非常にたくさんのフォントが利用できます。

残念ながら、この形式を (訳注: FreeBSD で) この形式を利用できるアプリケーションは、筆者が知る限りほとんどなく、 Ghostscript と Povray しか思いつきません。ドキュメントによれば、Ghostscript の 対応は不十分であり、タイプ 1

フォントより粗悪な結果になるようです。 Povray バージョン 3 もまた TrueType フォントを利用可能ですが、筆者は、ドキュメントを一連のレイトレースしたページとして作成する人が多いのではないかと疑っています :-)。

このなんとも悲惨な状況は変わりつつあります。 FreeType プロジェクト では FreeType の便利なツールを開発しています。

- XFree86 4.x に含まれている freetype モジュール。 詳細は FreeBSD ハンドブック か XFree86 4.0.2 Fonts ページを見てください。
- X11 用の xfsft フォントサーバは 一般のフォントに加えて TrueType フォントを提供します。 現在ベータ版であるにもかかわらずたいへん評判がいいものです。 詳しくは Juliusz Chroboczek's page をごらんください。 FreeBSD への移植についての情報は Stephen Montgomery's software page にあります。
- xfstt は X11 用のもうひとつのフォントサーバで、 link: ftp://sunsite.unc.edu/pub/Linux/X11/fonts/ から入手できます。
- `ttf2bdf` というプログラムは、 X の環境下で TrueType フォントのセットから BDF 形式のファイルを作るものです。 Linux 用のバイナリが ftp://crl.nmsu.edu/CLR/multiling/General/ から 入手できます。
- そしてその他 ...

10. どこでフォントを入手できますか?

インターネット上でたくさんのフォントを利用することができます。
これらは完全に無料であるか、シェアウェアです。加えて、
たくさんのフォントが収録されたあまり高価ではない CDROM がたくさんあります。インターネットでのアクセスポイント (1996年8月現在)を以下に示します。

- http://www.simtel.net/
- http://www.freshmeat.net/
- Ports Collection の x11-fonts/

11. その他の質問

- .pfm ファイルを利用するものはあるのか?
- .afm ファイルを .pfa もしくは .pfb から作成できるか?
- 非標準キャラクタ名がある PostScript フォントを Groff キャラクタにマッピングするファイルをどのように作成するか?
- xditview と devX?? デバイスで新たなファイル全てにアクセスするためのセットアップをすることができるか?
- Povray と Ghostscript で TrueType フォントを利用する例があるといいだろう。