TrueType Fonts in Debian mini-HOWTO

Bear Giles, *bgiles@coyotesong.com* v0.3, 10 gennaio 2000

Questo documento descrive come configurare un sistema Debian per usare i font TrueType per la visualizzazione e la stampa. La versione più recente di questo documento può essere ottenuta in formato HTML a *http://www.dimensional.com/~bgiles/debian-tt.html* . Traduzione a cura di Giovanni Bortolozzo, *borto (at) pluto.linux.it* .

Indice

1 Introduzione

Una installazione di Linux tipicamente contiene parecchi insiemi indipendenti di font e metriche di font. Un'occhiata veloce al mio sistema rivela font e metriche di font sparse nelle seguenti directory:

- `XFree86` conserva i suoi font in `/usr/X11R6/lib/X11/fonts/`;
- `Ghostscript` li conserva in `/usr/lib/ghostscript/fonts/`;
- `TeX` li tiene in `/usr/lib/texmf/fonts/`;
- Il pacchetto `kbd` di Debian mette i suoi font in `/usr/share/consolefonts/`;
- `Groff` conserva le sue metriche in `/usr/share/font/devps/`;
- `Enscript` (convertitore da ASCII a PostScript) tiene le sue metriche in `/usr/share/enscript/*.afm`.

Neanche a dirlo, questi font non sono coordinati.

I font TrueType sono stati progettati per eliminare questo problema permettendo che gli stessi file di font siano usati sia per la visualizzazione che per la stampa. Questo Howto prova a mostrare come usare questi font TrueType per la visualizzazione, la stampa e la preparazione di documenti su sistemi Debian GNU/Linux.

1.1 Liberatoria

Le informazioni in questo documento sono, al meglio delle mie conoscenze, corrette. Comunque questo HOW-TO è ancora in versione preliminare e quel che funziona per me potrebbe non funzionare per qualcun'altro. Anche se funziona, non sono uno scrittore tecnico professionista e sono noto per sorvolare sui dettagli critici.

Quindi divertitevi, ma giocando sicuri e facendo dei backup.

1.2 Crediti

Creato da Bear Giles, <*bgiles@coyotesong.com*>

I miei ringraziamenti vanno a:

- Brion Vibber, <*brion@pobox.com*> , che ha scritto il *(preliminare) TrueType HOW-TO (http://pobox.com/~brion/linux/TrueType-HOWTO.html)*
- Doug Holland, <*meldroc@frii.com*> , che ha scritto l'*XFree86 Font Deuglification HOW-TO (http://www.frii.com/~meldroc/Font-Deuglification.html)*

1.3 Link addizionali

Anche se questi link non discutono direttamente problematiche relative ai sistemi o ai pacchetti Debian, possono comunque risultare interessanti per i lettori di questo mini-HOWTO.

- Using TrueType Fonts with RedHat Linux *(http://www.kegel.com/linux/tt.html)*
- X Font Tools *(http://www.hex.net/~cbbrowne/xfonts.html)*
- Getting fonts to look pretty under GNU/Linux for applications like StarOffice *(http://king.ccrc.wustl.edu/~zubin/fonts.html)*
- How to use True Type © fonts for StarOffice Under Linux *(http://www.mindspring.net/~john_mcl/adding_fonts.html)*

1.4 Storia delle revisioni

- 0.1. 16 set. 1999: prima versione.
- 0.2. 16 set. 1999: aggiunta la sezione «Link addizionali».
- 0.3. 10 gen. 2000: commenti dei lettori! Aggiunti chiarimenti su diversi dettagli.

1.5 Commenti

Commenti, correzioni, aggiunte e critiche sono sempre benvenuti. Mi si può raggiungere a *bgiles@coyotesong.com* .

1.6 Progetti correlati

- FreeType

  http://www.freetype.org/ `http://www.freetype.org/`

  Il motore FreeType è un motore libero e portabile per il rendering TrueType. Il codice è una buona implementazione totalmente indipendente da quelle di Apple o Microsoft (comunque recentemente è sorto un problema riguardo un possibile *brevetto* della Apple). FreeType è una libreria non un server per i font o una libreria completa per il rendering del testo.

1.7 Distribuzione

Questa è la prima stesura e mi aspetto di cambiarla significativamente dopo la sua pubblicazione. Invito a fare riferimento all'ultima versione a *http://www.dimensional.com/~bgiles/debian-tt.html* . La home permanente per questo documento alla fine sarà a *http://www.coyotesong.com/*

2 Schermi (quel che non vi hanno mai detto)

Prima di tuffarci nell'impostazione dei font TrueType sotto X, rivediamo la differenza tra punti e pixel... e perché ce ne preoccupiamo tanto.

Tutti i font visualizzati sono misurati in *punti*. Un pollice è esattamente 72 punti. Perché 72? In parte a causa delle limitazioni delle macchine tipografiche meccaniche e in parte perché è sempre divisibile per 2, 3, 4, 6, 8, 9, 12, 18 e 24. Non è male notare che l'unità predefinita in PostScript è un punto.

(Nota storica: Ho mentito. Prima dell'introduzione di PostScript un pollice era esattamente 72,27 punti, ma tale dimensione del punto era stata impostata nell'epoca delle stampatrici meccaniche con stampi in metallo. Con gli schermi dei computer e le stampanti laser è semplice ottenere font di qualsiasi dimensione e quindi 72 ha *molto* più senso per le ragioni suddette.)

Come regola generale, la maggior parte del testo dovrebbe essere tra i 7 e i 12 punti. Qualsiasi cosa più piccola di 6 punti è detto letteralmente «fine print». Le stampanti ad aghi usano caratteri a 9 o 12 punti (rispettivamente per 8 o 6 righe/pollice).

Viceversa, tutti i driver video devono misurare i font usando il *pixel* come unità fondamentale. Per il proprio driver video, il proprio schermo è 1024x800 pixel, non 10 per 8 pollici (o 720 per 576 punti).

Per passare da punti (usati per specificare la dimensione dei font) a pixel (usati per i riferimenti nella memoria video) si deve conoscere la risoluzione del proprio schermo. Questa è solitamente misurata in «punti per pollice» (dpi) (Ndt: «point» sono i punti usati per misurare i font, mentre questi punti sono «dot»), sebbene siano in realtà pixel per pollice. Questa è l'unità di misura utilizzata in due insiemi di font bitmap inclusi con XFree86: l'insieme `fonts-75` è pensato è per l'uso con schermi di fascia bassa con una risoluzione di circa 75 dpi, mentre `fonts-100` è pensato per schermi di fasci media con una risoluzione di circa 100 dpi. Non ci sono font bitmap per l'uso con schermi di fascia elevata con risoluzione oltre i 120 dpi.

Come esempio concreto, uno schermo di 13 di diagonale (11.1 utilizzabili) che visualizza un'immagine di 640x480 pixel ha una risoluzione di 72.0 dpi. Questa Non È Una Coincidenza. Infatti, la maggior parte delle pagine web (e applicazioni Microsoft) sono progettate attorno ad uno schermo canonico con una risoluzione di esattamente 72 dpi. La configurazione predefinita di XFree86 assume uno schermo con una risoluzione di 75 dpi.

Tornando al mondo reale, nessuno usa più i video a 640x480, così come nessuno usa più schermi con 13 di diagonale. Poiché le schede video sono migliorate più velocemente dei monitor non è strano avere una configurazione come la mia: schermo con 19 di diagonale (più di 17 usabili), 1600x1200 pixel, risoluzione di 117 dpi.

Se utilizzo una configurazione standard di X, tutti i miei font sono approssimativamente 2/3 delle dimensione voluta. Non è un'esagerazione dire che tutti i font sembrano ridotti di «un'ordine di grandezza»: i font più grandi (12 punti) sembrano quelli medi (9 punti), mentre i font medi (9 punti) sembrano quelli piccoli (6 punti).

Si possono fare tre cose per correggere questo problema. Per prima cosa si dovrebbe dire al server X la risoluzione reale del proprio schermo:

/etc/X11/xdm/Xservers

```
#:0 local /usr/X11R6/bin/X -bpp 16
:0 local /usr/X11R6/bin/X -bpp 16 -dpi 120
```

Poi, ci si deve assicurare di usare come font bitmap preferenziali quelli a 100 dpi piuttosto di quelli a 75 dpi:

/etc/X11/XF86Config

```
Section "Files"
    RgbPath     "/usr/X11R6/lib/X11/rgb"
    FontPath    "/usr/X11R6/lib/X11/fonts/100dpi/"
    FontPath    "/usr/X11R6/lib/X11/fonts/75dpi/"
    FontPath    "/usr/X11R6/lib/X11/fonts/misc/"
    FontPath    "/usr/X11R6/lib/X11/fonts/Type1/"
    FontPath    "/usr/X11R6/lib/X11/fonts/Speedo/"
EndSection
```

Per finire, se il server X non trova un font bitmap che corrisponde esattamente a quanto richiede, proverà a «scalare» un font simile replicando i pixel («pixel replication»). Raramente i risultati sono piacevoli da usare. Assumendo di possedere un sistema ragionevolmente potente, possiamo dire al server di usare come prima scelta i font scalati, poi i bitmap per le corrispondenze esatte e come ultima risorsa i bitmap scalati.

/etc/X11/XF86Config

```
Section "Files"
    RgbPath     "/usr/X11R6/lib/X11/rgb"
    FontPath    "/usr/X11R6/lib/X11/fonts/Type1/"
    FontPath    "/usr/X11R6/lib/X11/fonts/Speedo/"
    FontPath    "/usr/X11R6/lib/X11/fonts/100dpi/:unscaled"
    FontPath    "/usr/X11R6/lib/X11/fonts/75dpi/:unscaled"
    FontPath    "/usr/X11R6/lib/X11/fonts/misc/"
    FontPath    "/usr/X11R6/lib/X11/fonts/100dpi/"
    FontPath    "/usr/X11R6/lib/X11/fonts/75dpi/"
EndSection
```

2.1 Uno sguardo al prossimo XFree86 4.0

Sono appena venuto a sapare che XFree86 supporterà le comunicazioni DDC, se sia il monitor che la scheda video le supportano. Ciò permetterà al server X di interrogare il monitor sulle sue reali dimensioni fisiche, permettendogli così di calcolare automaticamente le impostazioni DPI corrette.

Si dovranno ancora fare le modifiche appropriate al proprio font path poiché al server non dovrebbe essere permesso ignorare le impostazioni esplicite dell'utente in base a ciò che ha scoperto.

2.2 Panoramica: Preparazione prima di modificare i file di configurazione di `X`

Ogni volta che si fanno delle modifiche ai file di configurazione di X11, è un'idea eccellente disabilitare `XDM` mettendo `exit 0` da qualche parte all'inizio del file. Se non lo si fa e `X` per qualche ragione non riesce a partire, `XDM` metterà il sistema in un brutto circolo vizioso dal quale è difficile uscire. Uomo Avvisato, Mezzo Salvato.

3 Visualizzazione (`X` e i font server)

A rigore, non è assolutamente necessario impostare dei font server per usare i font TrueType con i server `X`. Se si vogliono usare file statici invece di un font server, si vedano le istruzioni per impostare i font TrueType per `ghostscript`.

3.1 Impostare il font server `xfs`

A questo punto assumo si abbia un file `/etc/X11/XF86Config` funzionante che carica le directory specificate esplicitamente nel `FontPath`. Lo si convertirà per usare il font server `xfs`.

- Installare `XFS` Se non lo si è già fatto, si installi `main/binary-*/x11/xfs_*.deb`.
- Configurare `XFS` Si editi `/etc/X11/xfs/config` e si cambi `catalogue` per contenere la propria `FontList`. Può essere utile anche cambiare il valore di `default-resolutions` (risoluzioni predefinite).

/etc/X11/xfs/config

```
# percorsi nei quali cercare i font
catalogue =
    /usr/X11R6/lib/X11/fonts/100dpi/:unscaled,
    /usr/X11R6/lib/X11/fonts/75dpi/:unscaled,
    /usr/X11R6/lib/X11/fonts/misc/,
    /usr/X11R6/lib/X11/fonts/Speedo/,
    /usr/X11R6/lib/X11/fonts/Type1/,
    /usr/X11R6/lib/X11/fonts/100dpi/,
    /usr/X11R6/lib/X11/fonts/75dpi/
# x1,y1,x2,y2,...
default-resolutions = 100,100,75,75
```

- Riavviare `XFS` Riavviare `XFS` nel modo solito di Debian:

shell di root

```
# /etc/init.d/xfs restart
```

- Verificare che `XFS` stia funzionando Prima di modificare il file `XF86Config`, si dovrebbe verificare che il server `xfs` stia funzionando elencando i font tramite il font server:

shell utente

```
$ fslsfonts -server unix/:7100
-adobe-courier-bold-i-normal--0-0-0-0-m-0-iso8859-1
-adobe-courier-bold-o-normal--0-0-100-100-m-0-iso8859-1
-adobe-courier-bold-o-normal--0-0-75-75-m-0-iso8859-1
....
```

- Modificare `/etc/X11/XF86Config` per usare `xfs` Ora si è pronti per dire al server X di usare il font server `xfs`. Si mantenga un font path statico come posizione di ripiego in caso di problemi con `xfs` (conviene mantenere «misc» poiché contiene `fixed`, il font predefinito).

/etc/X11/XF86Config

```
Section "Files"
    FontPath   "unix/:7100"
    FontPath   "/usr/X11R6/lib/X11/fonts/misc/"
EndSection
```

- Riavviare `X` Riavviare `X` con `startx`, a meno che non si viva pericolosamente. Se così è, mi si nomini beneficiario della propria polizza vita inusualmente grande e si riavvii `xdm`.

- Verificare che i font di `XFS` siano visibili Una volta stabilita una sessione X, si può verificare che il proprio server veda tutti i font di `xfs` elencandoli attraverso il server `X`.

shell utente

```
$ xlsfonts
-adobe-courier-bold-i-normal--0-0-0-0-m-0-iso8859-1
-adobe-courier-bold-o-normal--0-0-100-100-m-0-iso8859-1
-adobe-courier-bold-o-normal--0-0-75-75-m-0-iso8859-1
....
```

3.2 Impostare il font server xfstt ed installare i font TrueType

A questo punto assumo che si abbia un font server `xfs` funzionante e che si voglia aggiungere il supporto TrueType attraverso `xfstt`.

- Installare XFSTT Se non lo si è già fatto, si installi `main/binary-*/x11/xfstt_*.deb`.
- Installare i font TrueType Si copino i propri font TrueType nella directory `/usr/share/fonts/truetype`. Questi file solitamente hanno l'estensione `.ttf` e dovrebbero avere i permessi 0444.
- Riavviare il font server XFSTT Si riavvii il server `xfstt` con l'opzione `force-reload`

shell di root

```
# /etc/init.d/xfstt force-reload
```

- Verificare che XFSTT stia funzionando Prima di modificare il nostro file `XF86Config`, si dovrebbe verificare che il server `xfstt` stia funzionando.

  Importante: il server `xfstt` della Debian è in ascolto sulla porta 7101 non sulla 7100. Inoltre i permessi predefiniti richiedono che si esegua questa interrogazione come root.

shell di un utente e di root

```
$ fslsfonts -server unix/:7101
_FSTransSocketUNIXConnect: Can't connect: errno = 111
fslsfonts:  unable to open server "unix/:7101"

# fslsfonts -server unix/:7101
-ttf-arial black-medium-r-normal-regular-0-0-0-0-p-0-iso8859-1
-ttf-arial mt black-medium-r-normal-regular-0-0-0-0-p-0-iso8859-1
-ttf-arial narrow-bold-i-normal-bold italic-0-0-0-0-p-0-iso8859-1
```

- Modificare `/etc/X11/XF86Config` per usare `xfstt` Ora si è pronti per dire all'X server di usare il font server `xfstt`. Vogliamo che preferibilmente usi i font TrueType piuttosto che tutti gli altri.

/etc/X11/XF86Config

```
Section "Files"
    FontPath   "unix/:7101"
    FontPath   "unix/:7100"
    FontPath   "/usr/X11R6/lib/X11/fonts/misc/"
EndSection
```

- Verificare che i font di XFSTT siano visibili Una volta stabilita una sessione X, possiamo verificare che il nostro server veda tutti i font TrueType elencandoli.

```
$ xlsfonts | grep ttf
-ttf-arial black-medium-r-normal-regular-0-0-0-0-p-0-iso8859-1
-ttf-arial mt black-medium-r-normal-regular-0-0-0-0-p-0-iso8859-1
-ttf-arial narrow-bold-i-normal-bold italic-0-0-0-0-p-0-iso8859-1
....
```

3.3 Usare i font TrueType

A questo punto dovrebbe essere possibile usare un font TrueType in applicazioni come `GIMP`, `Netscape` o `StarOffice`. Poiché la maggior parte dei font TrueType non sono «equispaziati» (monospaced) probabilmente non conviene usarli con `xterm`.

3.4 Installare font TrueType addizionali

Se si sta usando il font server `xfstt`, è banale installare font TrueType addizionali.

- Copiare i nuovi font in `/usr/share/fonts/truetype/`
- Riavviare `xfs` con `/etc/init.d/xfs restart`

3.5 Internazionalizzazione

`xfstt` ha la capacità di generare diverse codifiche di font, a patto che i font TrueType contengano i glifi (glyph) necessari. Per abilitare font diversi da iso8859-1/unicode-1, si deve modificare manualmente lo script `/etc/init.d/xfstt`:

/etc/init.d/xfstt

```
- start-stop-daemon --start --quiet --exec $XFSTT -- \
      --port $portno --daemon
+ start-stop-daemon --start --quiet --exec $XFSTT -- \
      --port $portno --encoding iso8859-1,koi8-r,windows-1252,symbol-0 \
      --daemon
```

Le codifiche riconosciute in Debian 2.1 sono:

- iso8859-1 (Latin 1 - Europa occidentale)
- iso8859-2 (Latin 2 - Europa centrale e orientale)
- iso8859-3 (Latin 3 - esperanto e maltese)
- iso8859-4 (Latin 4 - rimpiazzata da Latin 6)
- iso8859-5 (cirillico)
- iso8859-6 (arabo)
- iso8859-7 (greco)
- iso8859-8 (ebraico)
- iso8859-9 (Latin 5 - Latin 1 con turco invece di islandese)
- iso8859-10 (Latin 6 - lingue nordiche, rimpiazza Latin 4)
- koi8-r (cirillico)
- windows-1250 (Europa centrale)
- windows-1251 (cirillico)
- windows-1252
- windows-1253 (greco)
- windows-1254
- windows-1255

- windows-1256
- windows-1257
- symbol-0
- wingding-0
- wingreek-0
- cp-437 (varie pagine di codice IBM)
- cp-737
- cp-850
- cp-851
- cp-852
- cp-853
- cp-855
- cp-857
- cp-860
- cp-861
- cp-862
- cp-863
- cp-864
- cp-865
- cp-866
- cp-869
- cp-895
- atari-st
- unicode-2

I primi 128 caratteri nelle codifiche `iso8859-x` sono sempre l'ASCII. I font `windows-` abbracciano e estendono gli `iso8859-1` con caratteri addizionali come «smart quotes» (poiché queste estensioni, come le «smart quotes» non sono definite in `iso8859-1`, sono solitamente rese con dei punti di domanda).

Eccellenti fonti di informazioni aggiuntiva sulle codifiche dei set di caratteri si possono trovare a

- *http://ppewww.ph.gla.ac.uk/%7Eflavell/iso8859/iso8859-pointers.html* ,
- *http://anubis.dkuug.dk/i18n/* e
- *http://czyborra.com/charsets/iso8859.html* .

3.6 Questioni di sicurezza

Qua sopra ho usato i socket Unix, ma i pacchetti Debian standard configurano anche `xfs` e `xfstt` per restare in ascolto sulle porte 7100 e 7101, rispettivamente. L'accesso a queste porte dovrebbe essere controllato dal campo `trusted-clients` in `/etc/X11/xfs/config`, ma questa opzione non è implementata in XFree86 3.3.2.3a.

Ciò significa che chiunque, da ovunque, può connettersi al font server. Poiché `xfs` (e presumibilmente `xfsts`) si «clonano» per supportare più utenti esiste un banale attacco tipo denial-of-service contro questi sistemi. Probabilmente è sicuro utilizzare i font server su linee in dialup (poiché si è presenti per gestire il problema), ma gli utenti di DSL e cable modem dovrebbero usare un firewall.

4 Stampa (ghostscript)

A partire dalla versione 4, `ghostscript` supporta i font TrueType con un'opzione in compilazione. Due pacchetti Debian forniscono `ghostscript`:

- `main/binary-*/text/gs_*.deb` versione 5.10 conforme alle DFSG,
- `non-free/binary-*/gs-aladdin_*.deb` versione 5.50 non conforme alle DFSG.

Entrambe le versioni supportano i font TrueType.

4.1 Configurare Ghostscript per usare i font TrueType

Se si ha un server `xfstt` funzionante, è facile configurare `ghostscript` per usare i font TrueType. Si esegua semplicemente il comando seguente:

```
# xfstt --gslist --sync >> /etc/gs.Fontmap
```

Nella pratica ho scoperto che trae vantaggio da alcune piccole modifiche alle definizioni dei font generate da `xfstt`. Per prima cosa, se un nome di font non contiene alcuno spazio, cambio il nome nella notazione usuale. Se un nome di font contiene degli spazi, rimpiazzo tutti gli spazi con trattini e il nome originale viene aggiunto come alias per il nuovo nome.

Per finire, antepongo `TTF-` (o `MS-`) a tutti i nomi dei font per minimizzare i problemi causati da font TrueType che hanno lo stesso nome di font preesistenti.

Quindi

```
(Arial)              (/usr/share/fonts/truetype/arial.ttf)   ;
(Arial Bold Italic)  (/usr/share/fonts/truetype/arialbi.ttf) ;
```

diventa

```
/MS-Arial             (/usr/share/fonts/truetype/arial.ttf)   ;
/MS-Arial-Bold-Italic (/usr/share/fonts/truetype/arialbi.ttf) ;
(Arial Bold Italic)   /MS-Arial-Bold-Italic                   ;
/Arial                /MS-Arial                               ;
```

Gli alias assicurano che `ghostscript` e `xfstt` possano ancora specificare lo stesso font con un nome comune.

Molto più importante, con le modifiche ai nomi dei font è possibile istruire `ghostscript` ad usare il font TrueType invece dei font standard. La documentazione afferma che questo è possibile anche con la notazione con le parentesi, ma io non sono riuscito a farlo funzionare.

Per esempio, possiamo istruire `ghostscript` a rimpiazzare i font Helvetica con i font Arial gratuiti della Microsoft aggiungendo le righe seguenti al file `/etc/gs.Fontmap`:

```
/Helvetica              /MS-Arial              ;
/Helvetica-Oblique      /MS-Arial-Italic       ;
/Helvetica-Bold         /MS-Arial-Bold         ;
/Helvetica-BoldOblique  /MS-Arial-Bold-Italic  ;
```

Alias simili possono essere definiti per gli altri font standard. Questi alias possono essere molto utili sulle stampanti `samba` che servono client Windows.

4.2 Stampare i campioni dei font TrueType

Il modo migliore per verificare che `ghostscript` sia configurato propriamente per usare i font TrueType è di stampare le pagine «campione» dei font. Assumendo di usare `ghostscript` 5.50 e che sia la propria coda di stampa predefinita, si possono stampare tutti i font TrueType con il seguente comando:

```
# xfstt --gslist --sync | printfont
```

dove `printfont` è il seguente script shell

```
#!/bin/sh

set -e
IFS= ')'

while read fontname rest
do
    cat << EOM | lpr
%!PS
(/usr/lib/ghostscript/5.50/prfont.ps) run
$fontname) DoFont
EOM
done
```

Se si vogliono stampare solo alcuni font, lo script seguente sarà più facile da usare:

```
#!/bin/sh

set -e
while read -p "Nome di font, o ^D per uscire: " fontname
do
    cat << EOM | lpr
%!PS
(/usr/lib/ghostscript/5.50/prfont.ps) run
$fontname DoFont
EOM
done
```

5 Generazione dei contenuti

5.1 Generare le metriche di font AFM

I file di metrica `AFM` non sono necessari per visualizzare file già esistenti con i font TrueType, ma sono necessari per creare nuovi file. Per generare questi file di metrica potrebbe essere usato il programma di ghostscript `/usr/lib/ghostscript/5.50/printafm.ps`, ma io ho trovato che il programma `ttf2afm` del pacchetto `tetex-bin` è più facile da usare.

Lo script seguente genererà un file `afm` per tutti i font TrueType in una directory:

```
#!/bin/sh

set -e

for i in *.TTF
do
    /usr/bin/ttf2afm $i > ${i%TTF}afm
done

for i in *.ttf
do
    /usr/bin/ttf2afm $i > ${i%ttf}afm
done
```

Un problema minore con `ttf2afm` è che alcune applicazioni si aspettano che i file `afm` comincino con il tag `StartFontMetrics`, mentre i file creati da `ttf2afm` iniziano con un commento. Questo «problema» è facilmente risolvibile modificando ciascun file con un editor di testo.

5.2 Generare i file `font.map`

Una volta che si hanno i file `afm`, si deve dire al sistema come trovarli. Questo spesso è fatto attraverso il file `font.map`.

Non mi è stato possibile trovare documentazione sul formato di questo file, diversamente da `fonts.dir`, `fonts.scale` e `fonts.alias`, tutti creati dal programma `mkfontdir`. Comunque il formato minimo sembra abbastanza semplice:

- Nome del font, senza spazi
- nome del file `AFM`, senza estensione

Gli alias sembrano essere implementati tramite voci multiple e l'estensione dei file deve essere in caratteri minuscoli.

5.3 Manipolazione di immagini: `GIMP`

`GIMP` è il programma di grafica e manipolazione di immagini della GNU. Non è servito fare modifiche ulteriori per usare i font TrueType con `gimp`.

5.4 Conversione da ASCII a PostScript: `enscript`

`Enscript` è un programma che converte ASCII in PostScript. Altri programmi che servono per scopi simili sono `a2ps` e `mpage`. Non riformatta il testo e solitamente è usato per stampare listati di codice sorgente.

Per usare i font TrueType con `enscript`, si devono fare due cose:

- Aggiungere `/usr/share/fonts/truetype` al proprio `AFMPath`.
- Specificare un font TrueType, esplicitamente o come alias di un font predefinito.

Per i dettagli si veda la documentazione di `enscript`.

Una volta fatte queste modifiche, non ho avuto problemi ad usare i font TrueType.

5.5 Formattazione e composizione di testi: groff

`Groff` è l'interfaccia Gnu al sistema di formattazione di documenti groff/troff. È con le pagine `man` che si può meglio apprezzare la potenza di `Groff`.

shell utente

```
$ zcat /usr/man/man1/groff.1.gz | groff -man | lpr
```

A parte le pagine man, una quantità incredibile di documentazione Unix usa la formattazione `troff` con le macro `ms` (e qualche volta `me`). Il pacchetto `xbooks` di Debian, per esempio, contiene 43 file che usano `troff` con le macro `ms`. Con `groff` questo materiale può essere stampato in maniera elegante.

`Groff` è un sistema molto potente, ma è il nipote (o il pronipote) di un programma utilizzato per la composizione dei testi per le presse di stampa degli anni '60. Il supporto dei font in `groff` riflette questo retaggio. `Groff`, diversamente dai suoi predecessori, usa PostScript come formato d'uscita predefinito e quindi il precedente lavoro fatto con `ghostscript` risolve metà del problema: `groff` non si deve preoccupare della lettura dei file dei font TrueType. Ha bisogno di avere a disposizione metriche di font accurate e questa sezione descrive come rigenerare i file di `groff` necessari:

File di descrizione PostScript di Groff

```
/usr/share/groff/font/devps/DESC
    File di descrizione del dispositivo
/usr/share/groff/font/devps/text.enc
    Codifica usata per i font testo
/usr/share/groff/font/devps/generate/textmap
    Mappa predefinita
/usr/share/groff/font/devps/generate/Makefile
    Makefile standard
```

Si deve modificare il Makefile,

/usr/share/groff/font/devps/generate/Makefile

```
- afmdir=/usr/local/afm
+ afmdir=/usr/share/fonts/truetype
```

cambiando il nome dei font con il loro equivalente TrueType (e.g., se si usano i font TrueType gratuiti di Microsoft si rimpiazzerà `Helvetica` con `Arial`), e modificare `TEXTFONTS` e simile per includere solo questi font ridefiniti.

Si deve pure modificare `/usr/share/groff/font/devps/generate/afmname` per usare i nomi dei font TrueType e i file `afm` e per rimuovere un'opzione `-e` da `awk`.

Dopo tutto questo, si possono ricostruire le tabelle di `groff` con

shell utente

```
$ cd /usr/share/groff/font/devps
$ make -f generate/Makefile
```

Come al solito, il modo migliore per verificare le modifiche è usare font facilmente riconoscibili. E.g. se si stanno usando i font TrueType gratuiti della Microsoft si può usare `Mistral` per `TR`.

(Il 1° aprile prossimo mi aspetto delle royalty da chiunque riconfigurerà il proprio sistema per stampare le pagine man con il font Old English!)

5.6 Formattazione e composizione di testi: `TeX`

`TeX` è un altro insieme molto comune di programmi per la formattazione e composizione del testo presente sulla maggior parte dei sistemi GNU/Linux.

I font di `TeX` possono essere creati con `mktexmf`, ma ho poche informazioni sull'esatto processo. Maggiori dettagli saranno forniti al più presto.

6 Domande non risposte

- Applicazioni La principale domanda senza risposta è perché se `netscape communicator` può usare i font TrueType, e `ghostscript` può usare gli stessi font TrueType, le pagine stampate da `netscape` non assomigliano a quelle sullo schermo.

  La risposta breve è che `netscape` genera l'output PostScript con i font standard (Helvetica e Times-Roman) invece di quelli specificati dall'utente o specificati nel file HTML. La risposta lunga è che non ho assolutamente idea del perché forzi questo comportamento o se ci sia un modo per evitarlo.

7 Ottenere i font TrueType

Se si cercano font TrueType ecco qui diversi posti dove si può cominciare a guardare:

- c:\windows\fonts sul fermaporte che c'è nell'angolo
- Microsoft's Free TrueType Fonts `http://www.microsoft.com/typography/fontpack/`
- Acid Fonts `http://www.acidcool.com/`
- The Font Ring `http://www.webring.org/cgi-bin/webring?ring=fontring&list`

7.1 Un commento sull'uso dei font TrueType gratuiti di Microsoft

Nessun articolo sul supporto TrueType sotto Linux potrebbe essere completo senza un commento sui font TrueType gratuiti di Microsoft. Vediamo prima se sia legale utilizzare questi font gratuiti di Microsoft:

`http://www.microsoft.com/typography/faq/faq8.htm`

```
Q. Cosa posso fare con questi font?
   . Chiunque può scaricare ed installare questi font per il
     proprio uso personale.
   . I progettisti [di pagine web] possono specificare questi
     font all'interno delle loro pagine web
   ...
```

Chiaramente, è legale e ragionevole per gli utenti Linux scaricare e installare questi font e quindi vorrei ringraziare la Microsoft per averli resi disponibili.

Poiché un'altra clausola restringe la loro redistribuzione «in qualsiasi forma che aggiunga valore a prodotti commerciali» non mi aspetto di vedere molto presto dentro a `main` il pacchetto con questi font (potrebbero essere impacchettati dentro a `non-free`, forse utilizzando un pacchetto installatore...?).

Ora che questo è stato chiarito, voglio dirigere la vostra attenzione sulla seconda clausola. Microsoft incoraggia attivamente gli sviluppatori di pagine web a specificare i suoi font nelle pagine web e molti editor HTML chiamano esplicitamente questi font.

Molti siti web, quando vengono visti con un Netscape/Linux standard, possono al più essere descritti come... decifrabili. Alcuni non lo sono nemmeno. Non è una coincidenza che i siti che sono, diciamo, leggibili sono solitamente quelli che specificano esplicitamente tutte le informazioni sui font nelle loro pagine web.

Dopo aver installato questi font ho notato che la maggior parte di questi siti problematici sono diventati significativamente più attraenti. Molti erano ancora altamente Windows-centrici, ma almeno non devo ritrarmi schifato ogni volta che carico le loro pagine.

La mia raccomandazione è di installare i font TrueType gratuiti della Microsoft per il proprio browser. Non è richiesto di creare pagine web usandoli e nemmeno di configurare `ghostscript` per usarli.

8 Copyright (in inglese)