

fonts: achtergrond

PostScript Fonts op computers?

Taco Hoekwater
taco.hoekwater@wkap.nl

abstract

Dit artikel geeft een korte inleiding in de interne werking van PostScript computer-fonts en hun coderingen. Dit artikel is een aanpassing van een serie slides die gepresenteerd werden op de ntg bijeenkomst van 22 oktober 1998 in Leuven.

keywords

fonts, computers

Inleiding

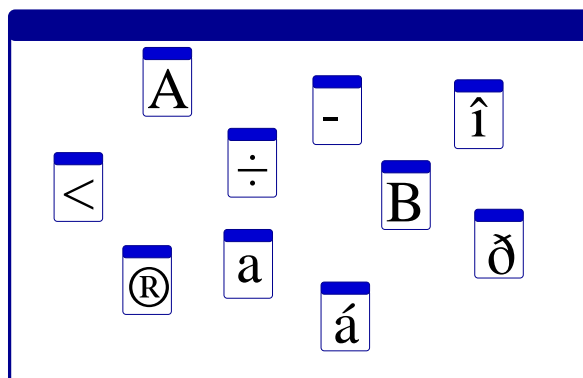
Een computer-font bestaat uit drie logische onderdelen:

- Algemene gegevens van het font
Hieronder vallen dingen zoals de naam van het font, en de ontwerp-grootte, maar ook zaken zoals kern-paren en hints voor lage-resolutie renderaars.
- Gegevens per karakter
Karakters hebben meestal zelf ook nog een naam, en die wordt hier bijgehouden. Tevens staan hier dingen zoals de breedte van het karakter.
- Een groep karakters
Van elk karakter staat hier aangegeven hoe het getekend moet worden. Dit kan per bestandstype verschillen: soms zijn het tekeninstructies, maar soms ook een bitmap plaatje of zelfs alleen een 'recept' om dit karakter te maken door bijvoorbeeld twee andere karakters over/naast elkaar te plakken.

Nu we deze algemene eigenschappen van een font gehad hebben, wordt het tijd om eens te gaan kijken hoe een en ander dan wordt opgeslagen op de harde schijf.

PostScript fonts

Een PostScript font onder Windows of Unix bestaat uit twee verschillende bestanden. De algemene gegevens en karakter-gegevens staan in een **afm** of **pfm** bestand. Voor Times-Roman heet dat bestand bijvoorbeeld **tir.....afm** (Bestanden die afkomstig zijn van Adobe Ssystems zijn vaak herkenbaar aan het feit dat Adobe de naam van een bestand aanvult met underscores, zodat het altijd precies acht letters in totaal zijn)



Figuur 1 Schematische opbouw

De karakters zelf staan in een **pfb** of **pfa** bestand. Let op dat zo'n bestand ook *in* een PostScript printer kan staan. In dat geval is het font niet beschikbaar op de harde schijf.

Voor het gebruik van een font binnen T_EX is alleen het bestand met de gegevens echt van belang. Het tweede bestand komt pas om de hoek kijken als er ook inderdaad wat moet worden afgedrukt, en valt daarom onder de verantwoording van de DVI driver.

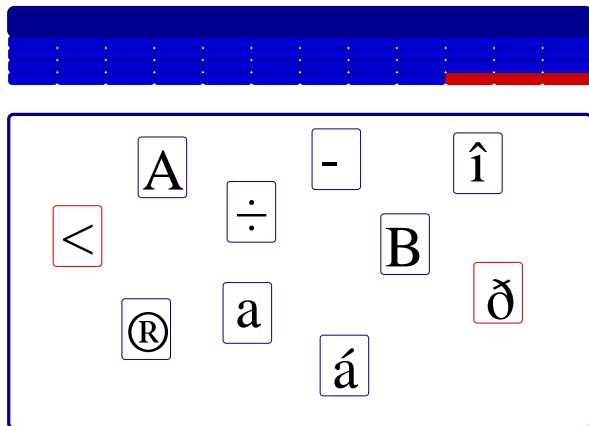
Het is bij het gebruik van een PostScript font handig om rekening te houden met het volgende:

- Er kunnen heel veel tekens in een font zitten.
- Daarvan zijn er maar 256 tegelijkertijd bereikbaar.
- Een **encoding** regelt welke dat zijn.
- Elk font heeft een ingebouwde encoding.
- Maar de kans is groot dat die encoding net niet die tekens bevat die je nodig hebt.
- Daarom is het mogelijk om een font te her-coderen, daarop kom ik later nog terug.

Hoe een driver een font vangt

Even aangenomen dat er een DVI beschikbaar is die PostScript fonts gebruikt en die je graag zou willen printen en/of bekijken. DVI's zijn onprintbaar, dus hier komt de DVI driver om de hoek kijken. In een DVI staan geen tekens, alleen gegevens.

Kortom: de driver moet op zoek naar het juiste font-bestand. Maar een driver weet niet zeker of een font al dan niet een PostScript font is (die informatie staat niet in de



Figuur 2 Een voorbeeld van een PS font met de ingebouwde encoding

DVI). Er is dus een routine nodig om te beslissen of een font PostScript is, of Metafont source, of ene virtueel font, of een TrueType font, enzovoorts.

Verreweg de belangrijkste driver als we het hebben over PostScript en -fonts is dvips. Dvips gebruikt een opzoek tabel om te kiezen tussen MetaFont (MF) en PostScript (PS) fonts. Die tabel wordt bewaard in een bestand met de naam **psfonts.map**. Deze methode is langzamerhand overgenomen door de meeste andere drivers (zelfs door 'nep'-drivers zoals **pdftex**, ook al heet het bestand dan anders).

De regel om te beslissen of een font al dan niet een PS font is is verbluffend eenvoudig:

- Als het font wordt genoemd in **psfonts.map** is het een PostScript font, in alle andere gevallen niet.

De opzet van psfonts.map

De vorm van het map-bestand is redelijk eenvoudig. Elke regel beschrijft precies n font, en die regel wordt verdeeld in een aantal velden die worden gescheiden door spaties.

Een voorbeeldregel is zoiets als:

```
tnr TimesNewRoman <tnr.pfb
```

De betekenis van de drie velden hierin is als volgt:

- tnr Dit is de T_EX naam van het font.
- TimesNewRoman De PS naam van het font. Die is nodig omdat PostScript interpreters de naam van een font nodig hebben om te kunnen switchen van het ene naar het andere font. De naam kan eventueel ook uit het .pfb bestand gehaald worden, maar dvips doet dat niet (nieuwere pdftex's doen dat wel).
- < betekent: voeg het volgende bestand toe aan de

PostScript uitvoer. We zullen later zien waarom het handig is om daar een apart karakter voor te hebben.

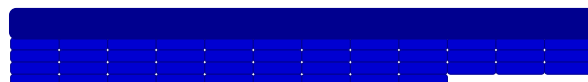
- tnr.pfb Het font zelf. Dit is meestal alleen een bestandsnaam, de driver zoekt gewoonlijk op een andere manier wel uit uit welke directory dat bestand moet komen.

Re-encoding

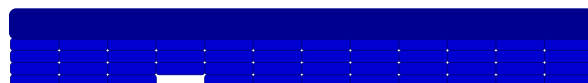
Het is mogelijk om een PostScript font anders te coderen met behulp van een **encoding**. Daarvoor is wat extra informatie nodig:

- Een bestand met de gewenste encoding, het .enc bestand.
- De naam van de encoding.

We nemen weer aan dat zo'n ge-hercodeerd font voor T_EX al bestaat: er is namelijk ook een andere TFM voor T_EX nodig (doordat sommige tekens nieuw of juist oud zijn kloppen de breedte in het 'standaard gecodeerde' TFM niet meer). Het verschil is grafisch voorgesteld in de twee figuren hieronder:



Figuur 3 Een voorbeeld van een TFM voor PS font met een ingebouwde encoding



Figuur 4 Een voorbeeld van een TFM voor PS font met een aangepaste encoding

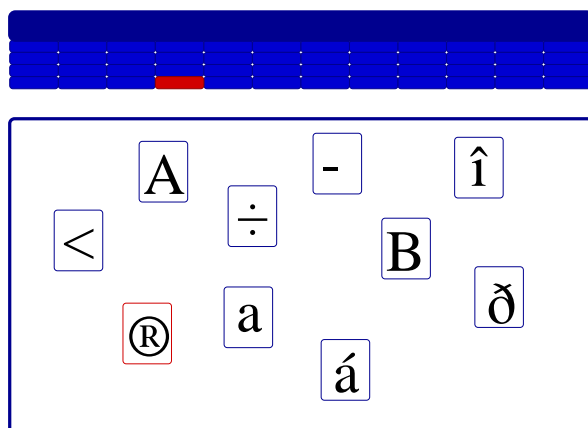
Voor **dvips** betekent de her-codering dan dat een extra regel moet worden toegevoegd aan het bestand **psfonts.map**, met bijvoorbeeld de volgende inhoud (de backslashes geven aan dat dit geheel op 1 regel hoort te staan):

```
tnr2 TimesNewRoman \
    "TeXnANSIEncoding ReEncodeFont" \
    <texnansi.enc \
    <tnr.pfb
tnr2 De TEX naam van het font. Deze naam is nu dus anders
TimesNewRoman
    De PS naam van het font. Maar deze naam verandert niet door het her-coderen, dus die blijft gewoon staan.
TeXnANSIEncoding
```

De naam van de encoding, zoals die gedefinieerd wordt in het `.enc` bestand.

`texnansi.enc` Het bestand waar de encoding in staat. Ook hier zou het mogelijk zijn om de naam door `dvips` uit het `.enc` bestand te halen, maar dit gebeurt niet (`pdftex` doet dit weer wel).

`tnr.pfb` Het font zelf, wat natuurlijk nog steeds precies hetzelfde bestand is. Immers: de tekens veranderen niet, ze worden alleen anders gerangschikt.



Figuur 5 Een voorbeeld van een PS font met een aangepaste encoding

Virtuele fonts

Virtuele fonts zijn weer een verhaal apart, en de bron voor een hoop extra complicaties. Het idee van een virtueel font is relatief eenvoudig: de gegevens die \TeX te zien krijgt in n `.tfm`, komen dan eigenlijk uit twee of meer ‘echte’ fonts. Een speciaal extra bestandje met als extensie `.vf` bevat dan de extra informatie die een driver nodig heeft om uit dat ene samengestelde font de verschillende onderdelen op te vissen.

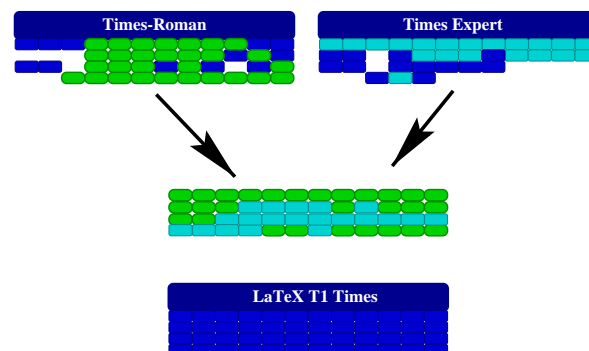
Een virtueel font gebruikt intern n of meer normale \TeX TFM bestanden als bron voor de gegevens. Deze fonts *kunnen* PostScript fonts zijn, maar dat hoeft niet. Deze fonts kunnen ook al ge-hercodeerd zijn, maar dat hoeft ook niet per se. Als laatste is het mogelijk om in een `.vf` bestand een ‘recept’ te bouwen voor het maken van een karakter (bijvoorbeeld een accent uit n font en een hoofd-karakter uit een ander).

Virtuele fonts worden in deze MAPS bijvoorbeeld gebruikt om de ‘mooie’ cijfers **12345** uit een ander font te halen. In het ‘normale’ Times font staan ‘saaie’ cijfers zoals 12345. Voordeel van deze aanpak als gebruiker is dat ik de vorige regel getypt heb als: cijfers 12345 uit ... zoals `12345`; dus zonder dat ik extra commando’s in

hoefde te typen.

Voor \TeX is een virtueel font absoluut niet anders dan een normaal font (wat logisch is, per slot van rekening heeft \TeX alleen maar de maten van het doosje om een teken nodig).

Maar dat gaat voor een DVI driver niet op, natuurlijk.



Figuur 6 Een virtueel font

Zoek de verschillen

Een DVI postprocessor zoals `dvips` gebruikt de volgende (alweer verdacht simpele) regel om te beslissen of een font virtueel is en er dus extra werk gedaan moet worden om de onderdelen bij elkaar te zoeken; of dat het een ‘normaal’ font is

- ▣ Als er een bestand met extensie `.vf` bestaat voor een font dat voorkomt in een DVI file *en* het komt niet voor in `psfonts.map`, dan is het een virtueel font.

`Dvips` gaat dus voor *alle* fonts die in de DVI genoemd worden *eerst* op zoek naar een bijpassende `.vf`. Als zo’n bestand bestaat, dan gaat `dvips` op zoek in die `.vf` om te ontdekken welke fonts dan wel gebruikt moeten worden. Vervolgens wordt voor de fontnamen die dan gevonden worden weer overnieuw besloten of ze al dan niet virtueel en/of PostScript zijn. Dat gaat zo door tot alle fonts gevonden zijn.

De belangrijkste regel die hieruit volgen is de volgende: Virtuele fonts horen *niet* in `psfonts.map` te staan. Als dat toch gebeurt gaat het gegarandeerd fout.

Mogelijke problemen met fonts

De regels hierboven zorgen ervoor dat vrij makkelijk kunnen controleren wat voor soort font-problemen bij wat voor soort fouten in de \TeX -installatie horen.

- ▣ Onvindbare bestanden van alle soorten
Er staat ergens een zoek-pad verkeerd.
Drivers moeten natuurlijk alle bestanden wel

correct kunnen vinden, gewoonlijk zoeken ze niet de hele harde schijf af (dat zou wel heel erg lang kunnen duren). Dit probleem laat zich oplossen door grondig de documentatie bij de \TeX -installatie te lezen en soms ook door bestanden te verplaatsen.

Een smerige truuk die erg vaak werkt: als je weet welk bestand er niet gevonden wordt, kun je dat bestand kopiëren naar de directory waarvandaan je de DVI driver aanroept. 10 tegen 1 dat het bestand dan wel gevonden wordt.

▣ Missende **tfm** bestanden

Dit gebeurt eigenlijk alleen met externe DVIs die je van iemand anders hebt gekregen. Als het toch een bestand van jezelf is, heb je een paar bestanden gewist sinds de compilatie, of het zoekpad voor de driver staat verkeerd (zoals hierboven), of je installatie is zo ernstig in de war dat alleen een expert of volledig her-installeren je nog kan redden.

▣ Missende **pfb** bestanden

Als **dvips** dit tegen je zegt, betekent dat dat het een regel heeft gevonden in **psfonts.map** voor dit font (anders zou het niet weten dat het een PS font was). Er zijn twee nieuwe mogelijkheden waarom deze fout kan ontstaan.

1) De schuldige kan **psfonts.map** zelf zijn, bijvoorbeeld omdat er een typefout in staat. Het is zelfs mogelijk dat die regel er helemaal niet in hoort te staan (zoals bij virtuele fonts, of wanneer een MF font toevallig dezelfde naam heeft als een PS font).

2) Het is een commercieel font wat je niet hebt, maar de \TeX distributie had er al wel **.tfm**'s voor meegestuurd. Distributies bevatten vaak als service allerlei bestanden voor fonts die gekocht moeten worden, vooral omdat het maken van de TFM en VF bestanden niet zo heel triviaal is. Deze goedbedoelde toevoeging kan wat misleidend zijn: commerciële fonts moeten nu eenmaal gekocht worden.

▣ Missende **.vf** bestanden

Deze fout is wat minder snel herkenbaar dan die hierboven. Je merkt dat er wat fout gaat omdat MetaFont wordt aangeroepen dan foutmeldingen begint te geven over niet gevonden **.mf** bestanden.

Dvips heeft voor zo'n font geen **.vf** gevonden, en ook geen regel in **psfonts.map**, en heeft daarom besloten dat dit dus een Metafont moet zijn.

▣ Missende regels in **psfonts.map**

De symptomen van deze fout zijn dezelfde als die van hierboven.

Voor \LaTeX gebruikers met een min of meer standaard systeem is meestal te achterhalen welke fout van deze twee het is door te kijken naar de naam van het bestand waar om gevraagd wordt bij Metafont.

Als die naam eindigt in **8a** of **8r** of begint met **rp** is het probleem waarschijnlijk dat de **psfonts.map** niet goed is.

Als de naam eindigt op **7t** of **8t** is er meestal een VF bestand zoekgeraakt.

▣ Verkeerde of missende encodings

Vanaf hier worden de problemen pas echt lastig. Deze fout en de fouten die hierna komen geven namelijk meestal *helemaal geen* foutmelding, maar zorgen er wel voor dat allerlei accenten verkeerd zijn en/of zelfs voor missende tekens in de uitvoer.

Het is in het algemeen erg lastig om te ontdekken welke van deze problemen de oorzaak is van de misre.

▣ Extra regels in **psfonts.map**

Regels voor virtuele fonts in **psfonts.map** zorgen er soms voor dat een VF onbedoeld verkeerd geïnterpreteerd wordt.

▣ Teveel TFM bestanden

Dit is vrij simpel, eigenlijk. Wat er gebeurt is dat \TeX en de driver verschillende TFM bestanden vinden, die niet helemaal precies hetzelfde zijn. De driver roept dan 'checksum error' of iets in die geest.

Zorg dat je altijd maar **n** TFM op je harde schijf hebt staan met dezelfde naam.

▣ Teveel VF bestanden

Je kunt inderdaad te veel VF bestanden hebben. Als **n** van de componenten van een virtueel font bedoeld is als 'gewoon' font, maar je hebt per ongeluk een bestand met dezelfde naam maar met extensie **.vf**, dan vindt **dvips** dat dit *dus* een virtueel font is en begint die **.vf** te lezen.

Mogelijke fouten kunnen van alles en nog wat zijn, dat hangt immers af van wat er in de **.vf** gedaan wordt. De meestvoorkomende oorzaak van dit probleem: er waren al allerlei bestanden meegeleverd voor het gebruiken van dit font (met de \TeX distributie) maar dat wist je niet en daarom heb je een font *nogmaals* geïnstalleerd op een ietwat andere plek.

▣ Mismatches tussen **tfm** en **vf** bestanden

Dit gebeurt vooral met DVI's die van iemand anders af komen. In dat geval is er eigenlijk niets aan te doen, behalve dan om een PostScript bestand vragen. Helaas is PostScript een stuk beter portable dan DVI, vooral als er virtuele bestanden gebruikt worden.

En nu?

Samenvattend kan er met fonts heel erg veel mis gaan. Een paar van de dingen hierboven zijn vrij makkelijk herkenbaar, en kunnen ook vrij simpel hersteld worden (bijvoorbeeld als je een PostScript font geïnstalleerd hebt) en je het

vergeten **psfonts.map** aan te passen.

Problemen met DVI bestanden die van iemand anders komen zijn vrijwel nooit op te lossen. Dan is het vrijwel altijd het best om PostScript of PDF bestanden te vragen in plaats van de DVI.

Andere problemen kunnen heel lastig zijn. Als je vermoedt dat het je eigen schuld is omdat je een stommitieit hebt begaan, is het vaak het best (en in ieder geval het snelst) om het hele \TeX systeem te herinstalleren. Als het probleem is ontstaan omdat je probeerde iets zelf uit te

breiden, is het waarschijnlijk het best om eerst te herinstalleren en dan nog eens te proberen.

En natuurlijk zijn er voor al dit soort gevallen de guru's van \TeX-NL . Blijf niet eeuwig proberen om iets op te lossen wat je eigenlijk niet duidelijk is. Als het over fonts gaat zijn zelfs de simpelste foutmeldingen soms gruwelijk ingewikkeld. Probeer een paar dingen en als het dan nog niet werkt, schroom dan niet op een vraag te posten naar \TeX-NL (als je geluk hebt kun je iemand opbellen, dat werkt vaak nog stukken beter).