

WISKUNDIGE GANZEVEREN

Dit artikel beschrijft de concepten die ten grondslag liggen aan kalligrafie met de computer. Als voorbeeld hiervan wordt het programma MetaFont gebruikt, dat al jaren oud is, maar zo goed is doordacht dat de principes nog steeds werkbaar zijn, en andere fontgeneratoren op vele punten het nakijken geeft!

Dit artikel is een voorbeeld bij het artikel ‘Modellieren — Professioneel Puzzelen’ uit de vorige I/O-Vivat. Verder is in die I/O-Vivat een artikel ‘Computer Typesetting — Constructies van Elastiekjes’ verschenen, dat aan dit artikel gerelateerd is: MetaFont wordt doorgaans namelijk gebruikt om de fonts te genereren voor  $\text{\TeX}$ .

GESCHIEDENIS

SPIJKERS MET KOPPEN

Kalligrafie is zo oud als de schrijfkunst. Het spijkerschrift van de Sumeriërs, de hiërogliefen van de Egyptenaren, de karakters van de Chinezen en de Gothische tekens zijn alle voorbeelden van symbolen die gebruikt werden om te schrijven. Maar natuurlijk ook het overbekende Times Roman, dat heden ten dage wordt gebruikt door een aantal vooraanstaande kranten.

Deze schriften zijn heel divers. Sommige zijn ontworpen voor religieuze doeleinden, andere voor administratieve aangelegenheden. En de richtingen waarin ze geschreven worden varieert ook, met als speciaal geval de Egyptische hiërogliefen, die in elke richting geschreven kunnen worden: horizontaal of vertikaal, en de hoofden in de hiërogliefen kijken naar het begin van de tekst! Dit was gewenst omdat de hiërogliefen vooral ter versiering dienden, en symmetrisch schrift werd door de wiskundig aangelegde Egyptenaren van die tijd fraai gevonden.

Toch hebben de schriften ook veel gemeenschappelijk. Bijna allemaal zijn ze ontstaan uit grafische afbeeldingen van dingen in de wereld. Om het schrijven niet te moeilijk te maken zijn ze grondig versimpeld, tot ze soms tot een paar

pennestreken teruggebracht waren.

Daarmee komen we op een interessant onderwerp: de pennestreken. In alle lettertekens is te herkennen met welk materiaal ze werden geschreven. Kijk maar:

**Spijkerschrift:** getekend door een puntige rieten stift zijdelings op een tablet natte klei te drukken. Het schrift bestaat uit langwerpige driehoekjes, vandaar haar naam.

**Hiërogliefen:** Geschilderd met penselen en inkt op waterbasis, of uitgehouwen in brokken steen van tweeduizend kilo. Het schrift wordt vaak verfraaid met diktevariaties.

**Chinese karakters:** Geschilderd met een vertikaal vastgehouden penseel. De streken lopen vaak van dik naar dun, waarbij de dikte los staat van de richting van de lijn.

**Gothische karakters:** Getekend met een ganzeveer, of tegenwoordig met een breedpuntige pen, die onder een hoek ten opzichte van de schrijflijn wordt gehouden. De verticale streken zijn dikker dan de horizontale, en rondingen lopen mooi glooiend over tussen die dikten.

**Times Roman:** Ook dit is een schrift zoals het Gothische, dat vertikaal dikker is dan horizontaal, door de zelfde oorzaak.

Gegeven deze achtergrond is het een niet geringe opgave om een stuk software te ontwikkelen dat het tekenen van fonts voor je doet. Toch integreert MetaFont al deze dingen. Maar er is meer...

EISEN VAN DE ZETTERS

BILLY MET DE KOP IN HET ZAND

Zoals al besproken is in mijn artikel over  $\text{\TeX}$ , kan een zetpakket al overweg met louter rechthoekjes. Een zetpakket zal zich niet druk maken over het plaatje dat in die rechthoekjes staat. Dat is de verantwoordelijkheid van fonttekenpakketten<sup>1</sup> zoals MetaFont. Maar dat

<sup>1</sup>Stelling: Nederlanders halen meer punten met Scrabble dan Amerikanen, die alleen met woorden van weinig lettergrepen overweg kunnen.

heeft enige implicaties.

Allereerst moeten de rechthoekjes zó worden doorgegeven, dat ze op een zelfde basisregel tegen elkaar aan geplakt kunnen worden. Dus als de afmetingen van een letter worden bepaald, moet niet alleen de hoogte worden gegeven, maar ook de hoeveelheid van die hoogte die onder de basislijn uitsteekt. Dat is nul voor een letter ‘x’, en meer dan nul voor een letter ‘g’. MetaFont geeft haar informatie aan T<sub>E</sub>X door als een drietal getallen: de breedte van een letter, de hoogte boven de basislijn, en de diepte onder de basislijn.

Verder wil een zetpakket graag weten of een letter een beetje mee kan rekken. Neem bijvoorbeeld een spatie tussen woorden in. Waarom zou die niet een beetje kunnen uitrekken of inkrimpen? Of, nog duidelijker, een spatie tussen twee zinnen (na een punt die een zin beëindigt). Dit soort informatie is fontafhankelijk. Denk maar eens aan een **Courier** font, waarin de spaties niet mogen rekken, en een willekeurig proportioneel font, waarin dat wel mag.

Nu is het zetpakket (meestal T<sub>E</sub>X als je over MetaFont praat) nog steeds niet tevreden. Want het stelt ook nog eisen aan gerelateerde fonts: die moeten dezelfde x-hoogte hebben. De x-hoogte is de hoogte van de letter ‘x’ (en de meeste andere letters), dus de afstand boven de basislijn waar de letters ophouden, behoudens eventuele ‘stokken’ of ‘krullen’. Dat dit echt nodig is zal duidelijk zijn uit het volgende voorbeeld:

xxx*xxxx*xxxxxxxxxxxxxxxx

Als dit met MetaFont en T<sub>E</sub>X is gedaan, kun je er een lineaal langs leggen, (behoudens ontwerpoverwegingen van de fontontwerper natuurlijk,) en zijn alle letters exact even hoog. Maar sommige programma’s, en ik noem bijvoorbeeld Word van ons aller Billy-Boy, voeren transformaties uit op de bitmap van de basisversie van het font, met als gevolg een afwijking van gangbare layout-wensen. Maar allez, commercieel gezien is dat niet nodig, blijkens Word. . .

Verder zal een zetpakket er van uit gaan dat voor *italics* een ander lettertype wordt gemaakt, vaak krulliger dan de romein (‘rechttop’) versie die er bij hoort. Dit in tegenstelling tot *slanted*, wat een kromgeduwde variant van romein is. (Joe hoe, Bill! Waar blijf je nou?)

Goed, er zijn dus ook nog aardig wat eisen vanuit de zethoek die gesteld worden aan een font-

tekenpakket. En toch... toch maakt MetaFont ze allemaal waar!

## PENNESTREKEN

... EN WISKUNDIGE STREKEN

MetaFont beschrijft pennestrekken. Zoals je in een kalligrafie-cursus wordt bijgebracht hoe je welke pen moet vasthouden, hoe je vloeiende krommes ermee moet trekken, hoe je de stokken en andere uitstulpingen van een letter tekent voor een bepaald font, zo leer je dat ook aan MetaFont.

En het leuke is dat je je veel werk kunt besparen, doordat MetaFont het woordje ‘meta’ in haar naam heeft: het is mogelijk om generiek te beschrijven hoe je een bepaald soort stok maakt, dus alle pennestrekken die je daarvoor moet maken, en vervolgens bij de ‘l’, de ‘k’ en de ‘b’ te zeggen dat ze allemaal diezelfde stok hebben. Sterker nog, als de ‘t’ een soortgelijke maar andere stok heeft, dan kun je je stokken zelfs geparameteriseerd definiëren, en een variant voor de ‘l’ en de ‘t’ genereren met dezelfde code, maar met een andere parameterwaarde.

Hoe ziet een pennestreek van MetaFont er uit? Wel, in het algemeen is dat een kromme. En die kromme gaat door een aantal punten heen. En hoe kun je krommen in het algemeen beschrijven? Dat gaat (in het 2-dimensionale geval dat voor layout geen beperking levert) prima met Bezier-krommen.

Bezier-krommen worden veel in grafische pakketten toegepast. Het zijn krommen die tussen twee punten worden getrokken, en wel zo dat er bij elk punt een hulppunt hoort. Als bij een punt  $p$  een hulppunt  $p'$  hoort, dan vertrekt de kromme vanuit  $p$  in de richting van  $p'$ , maar buigt daarna mogelijk af. Als het punt  $p'$  verder van  $p$  verwijderd ligt, zal die afbuiging echter langer duren.

MetaFont laat je toe om een aantal punten op te geven, al dan niet cyclisch afgesloten, waartussen dan Bezier-krommen worden getrokken. Dit soort constructie heet een ‘pad’ en MetaFont zal in principe zelf de hulppunten voor de Bezier-krommen bepalen. Daarbij geldt een aantal regels, met bijbehorende overrulingsmogelijkheden.

Ten eerste zullen twee Bezier-krommen aan weerszijden van een punt (‘vertrekkend’ en ‘aankomend’) dezelfde richtingscoëfficiënt hebben in dat punt, zodat de kromme vloeiend

door dat punt heen loopt. Dit kun je voorkomen door expliciet aan te geven dat het een hoekpunt betreft.

Ten tweede zullen de hulppunten zodanig worden gekozen, dat de resulterende krommen van het *gehele* pad zo vloeiend mogelijk worden gekozen. Dit is te overrulen door expliciet aan te geven dat een bepaalde Bezier-kromme wat meer ‘tension’ aan kan.

Het is mogelijk extra wensen op te geven bij de hulppunten. De eerste mogelijkheid is om alleen de richting van dat punt aan te geven, ofwel de richtingscoëfficiënt te geven in een punt. De tweede mogelijkheid is om met de hand aan te geven waar de hulppunten liggen. De derde mogelijkheid is om een punt te geven waardoorheen de kromme moet lopen, en wel met als richtingscoëfficiënt de richting van het beginpunt van die Bezier-kromme tot het eindpunt daarvan.

Mocht je de draad kwijt zijn, dan vat ik het bovenstaande samen door te zeggen dat je alle denkbare krommen heel goed kunt benaderen, en dat MetaFont zelf vaak de goede kromme tussen punten weet te vinden, maar dat je altijd precies kunt aangeven hoe je wensen anders zijn dan wat MetaFont als default-gedrag vertoont.

## PENPUNTEN

### DE PUNTJES OP DE ï

Terug naar de pennestroken, want dat onderwerp is erg interessant. Om te beginnen stelt MetaFont dat een pen een bepaald soort punt heeft. Om precies te zijn, je kunt aangeven welke vorm je aan de punt wilt hebben. Een ellips onder 30ø met de horizontale lijn? Of misschien een vierkante punt? De eerste geeft rondere vormen dan de tweede. Dit zijn dingen die je het duidelijkst kunt zien bij de hoeken van een letter.

Als voorbeelden: **Courier** teken je met een ronde punt (want de letters zijn overal even dik, vermoedelijk doordat de mallen waarin de loden drukletters werden gegoten gegraveerd werden met een ronde stift). Times Roman daarentegen, wordt getekend met een brede, bijna platte stift (zeer vlakke ovaal) onder 5ø tot 30ø met de horizontaal.

Dus als je een kromme lijn trekt met zo’n pen, dan zie je variërende diktes van lijnen, maar wel consistent door het hele schrift heen. Dit is heel gebruikelijk, met uitzondering van misschien een krulletje hier en daar, waarvoor je dan even een

andere pen pakt, zowel uit je kalligrafieset als uit je set van pennen binnen MetaFont. Ziehier, hoe nauwkeurig de kalligrafie-wereld in MetaFont is gemodelleerd.

Nu zijn er echter al enige problemen. Soms wil je een font maken dat niet bestaat uit het trekken van lijnen met een dikke pen, maar wil je een ingevuld vlak tekenen. Dit doe je met MetaFont door het vlak te omlijnen met een pen met een punt met dimensies nul, en vervolgens op te dragen dat vlak in te vullen (of juist te wissen). Daar is dus aan gedacht. Ook is het mogelijk (schijnt) om de druk op de pen te variëren gedurende het trekken van de lijn, wat resulteert in een variabele lijndikte, ook voor rechte lijnen. Of, je kunt een draaiing van de penpunt tijdens het trekken van een lijn opgeven, met als gevolg bijvoorbeeld een spijkerschrift-achtig effect. Wederom geef je op hoe de tekenaar werkt, niet wat de computer moet doen.

Stel nu, dat je een font wilt ontwerpen waarin je dezelfde vorm wilt gebruiken voor de normale en de boldface uitvoering. Veel tekstverwerkers schuiven hiertoe een letter wat heen en weer en op en neer, en nemen de **or** van de bitmaps die bij al die letters horen, en noemen dat boldface. Deze transformatie is een technische oplossing, en klopt niet met wat gangbaar is in de kalligrafie.

Dit is een rampzalige aanpak. Als de originele letter scherpe hoekjes had (b.v. door een rechthoekige penpunt) dan zijn die vermoedelijk verloren gegaan. En, openingen in je font zijn misschien wel verdwenen. En, misschien is het wel mooier om voor boldface een bredere pen te nemen, die even hoog is als die voor de normale letter. En, je wilt toch wel graag dezelfde x-hoogte voor boldfaced letters hebben als voor gewone letters, zelfs al weet iedereen dat boldface tekst in normale tekst niet aan te raden is. Dit zijn details, toegegeven. Maar met MetaFont worden deze dingen wel heel elegant opgelost: je kunt je font bijvoorbeeld parameteriseren met de pen die je wilt gebruiken. Dan kan het boldface lettertype met een andere pen worden getekend dan het normale lettertype. Het probleem met de x-hoogte wordt als volgt opgelost in MetaFont: Aangezien de vorm van een pen goed bekend is, kan de uitwijking naar boven ten opzichte van het middelpunt worden berekend (en dito naar links, rechts en onder). Deze kennis kan worden gebruikt door niet te specificeren dat de pen *door* een bepaald punt gaat, maar daar juist aan *raakt* met de

bovenkant. Dus de kromme beschrijft niet ‘de pen gaat door punt  $z$ ’, maar ‘de bovenkant van de pen raakt aan punt  $z$ ’. Elegant, nietwaar?

## TRANSFORMATIES

TREKKEN, DUWEN, STOTEN

Een willekeurige (pad)expressie in MetaFont kan lineair worden getransformeerd. Met ‘linear’ bedoel ik: je mag er aan trekken, duwen en buigen zoveel je wilt, zolang het maar overal even erg gebeurt. De volgende transformaties vallen hier binnen:

**translatie:** geef een offset in x- en y-richting.

**schaling:** geef een vermenigvuldigingsfactor in x- en/of y-richting.

**slant:** geef een hoek waaronder de informatie schuin moet worden gedrukt.

**roteer:** geef een hoek waarover de informatie ronddraait.

Deze mooi volledige set van transformaties maakt het bijvoorbeeld mogelijk om het lettertype SMALL CAPITALS handig te definiëren. Dit is een lettertype waarbij de kleine letters dezelfde vorm hebben als de hoofdletters, maar wel kleiner van afmetingen zijn. Bij sommige tekstverwerkers ziet dit er lelijk uit: De grote letters (ingetikt als hoofdletters) zijn vetter dan de kleine letters (ingetikt als kleine letters). Dit komt doordat de letters gewoon lineair worden verkleind. Wat je met MetaFont zou doen is het verkleinen van de paden waarlangs de pen zich beweegt, maar dan zouden de kleine letters wél met de zelfde pen worden getrokken als de hoofdletters, alleen langs een kleinere contour.

## SOFTWARE

GOED, BUGFREE EN GRATIS

MetaFont is een lastige taal, en als je eenmaal begint te lezen en denken over deze principes zul je merken dat je gaat gruwelen van heel veel drukwerk dat al snel als prutswerk door de mand valt. De producten die MicroSoft-Word opleveren zijn hier het (mij) bekendste voorbeeld van. Er zijn verscheidene omzettingen mogelijk met MetaFont output. Eén soort output van MetaFont is een font metrics (.tfm) file die bedoeld is voor T<sub>E</sub>X maar die ook voor andere pakketten

werkbaar is of kan zijn. Een ander soort output van MetaFont is de generic font (.gf) file, die een font in bitmapped formaat uitdrukt. Die kan met tools worden omgezet in device independent (.dvi) formaat en in packed font (.pk) formaat. Mogelijk zijn er meer tools, ik ken ze echter niet.

Ik heb weinig fontgeneratiepakketten gezien, maar wel heel wat fonts, en ik heb nog geen andere dan die van MetaFont gezien die aan mijn —hoge?— eisen voldeden. Daarom, als je overweegt om serieus met zetwerk bezig te gaan, overweeg dan ook of T<sub>E</sub>X niet iets voor je is. Zo niet, zoek dan absoluut uit of je in het pakket van jouw keuze gebruik te maken valt van fonts die door MetaFont zijn gegenereerd.

## REFERENTIES

VOOR LEESGIERIGE AAGJES

- Donald Knuth, *The MetaFont book*, alles wat je ooit wilde weten over MetaFont, en vermoedelijk nog veel meer.
- Anne Millard et al, *Geschiedenis voor de Jeugd — van het stenen tijdperk tot de val van Rome*, met veel details over de ontwikkeling van diverse schriftsystemen.
- Spectrum uitgeverij, *Dit is je wereld — een verkenning van alles wat we weten*, en daarin *De geschiedenis van de boekdrukkunst*.
- Gail en Christopher Lawther, *Leer kalligraferen, stap voor stap*, over kalligrafische technieken en font-ontwerp.
- W.V. Davies, *Reading the past: Egyptian Hieroglyphes*, over de letters, woordenschat en grammatica van het (midden) Egyptische hiërogliefenschrift.
- Donald Knuth, *The T<sub>E</sub>X book*, alles wat je ooit wilde weten over T<sub>E</sub>X, en vermoedelijk nog veel meer.
- Rick van Rein, *Computer Typesetting — Constructies met Elastiekjes*, over de principes achter T<sub>E</sub>X.

*Rick van Rein*

