

PostScript en \LaTeX , de komplementariteit in praktijk

Michel Goossens

CERN, CN Division
CH121 Geneva 23, Switzerland
goossens@cernvm.cern.ch

Abstract

In dit artikel toon ik aan hoe PostScript en \LaTeX een hoge graad van samenhang bezitten, die het mogelijk maakt om de voordelen van beide systemen te combineren om documenten elektronisch te publiceren. Allereerst vertel ik hoe, samen met de dvi-vertaler `dvips` en het stijlbestand `epsfig`, het invoegen van PostScript materiaal in een (\LaTeX) \TeX bestand heel eenvoudig wordt. Samen met de stijl `rotating` kan men bijna elk gewenst globaal grafisch effect verkrijgen zonder per-se een PostScript guru te zijn. In het tweede gedeelte van het artikel geef ik een kort overzicht van enkele op PostScript gebaseerde stijlbestanden, die bepaalde nuttige visuele effecten genereren, zoals grijze raampjes, kleurentypografie en het overdrukken van tekst. In het laatste gedeelte toon ik hoe eenvoudig het is om PostScript fonts te gebruiken in \LaTeX met het nieuwe fontselectie systeem (NFSS) van Frank Mittelbach.

Een woord voaaf

Dit artikel is een verkorte uit het Engels vertaalde versie van een hoofdstuk uit *het boek 'A \LaTeX Companion'*, dat ik samen met Frank Mittelbach (de leider van het \LaTeX 3 project) en Alexander Samarin (ISO, Internationale Standaards Organisatie, Genève), heb geschreven, en dat in september door Addison-Wesley wordt uitgegeven. *Dit boek zal in ongeveer 400 bladzijden een overzicht geven van al wat een gevorderd \LaTeX gebruiker nodig heeft om haar/zijn \LaTeX document in de gewenste structuur en vorm te gieten.* Kort samengevat ziet de inhoudstabel er als volgt uit: na een kort overzicht van wat \LaTeX (wel en niet) is behandelen we de structurele basisbegrippen (hoofdstukken en paragrafen, lijsten, verbatim omgevingen, verschillende vormen van noten, meerdere kolommen, de kop- en voetteksten, drijvend materiaal). Een volledig hoofdstuk is gewijd aan tabellen en aan het nieuwe fontselectie systeem (NFSS) waarvan weldra de tweede, erg uitgebreide versie, uitkomt. Vervolgens bespreken we de mogelijkheden op het gebied van wiskundig tijpzetten met $\mathcal{A}\mathcal{M}\mathcal{S}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ en behandelen het probleem van grafisch materiaal, eerst op een drukker-onafhankelijke manier (`picture` omgeving en de `epic` en `eepic` stijlen) en dan met PostScript (zoals in dit artikel). De volgende hoofdstukken bevatten een gedetailleerde bespreking van hoe men een zakenregister (`index`) genereert met `MakeIndex` en hoe men literatuurlijsten en bibliografische gegevensbanken creëert en beheert met `BibTeX`. Het laatste hoofdstuk vertelt hoe men zijn \LaTeX stijlen kan documenteren en verspreiden met de stijl `doc` en `docstrip`. De eerste appendix beschrijft waar

men terecht kan op het internet of via elektronische post om \LaTeX stijlbestanden en informatie te verkrijgen, terwijl de tweede appendix een gedetailleerd overzicht geeft van \LaTeX 's gevorderde begrippen (het definiëren van opdrachten en omgevingen, het gebruik van tellers en lengtes, de constructie en het nut van raampjes en tenslotte hoe men met \LaTeX arithmetiek bedrijft).

De helft van de opbrengst van dit boek gaat direct naar de financiering van het \LaTeX 3 project, zodat U, door het kopen van dit boek, zich niet alleen een handig referentiewerk aanschafft, maar ook ervoor zorgt dat \LaTeX in de toekomst nóg beter wordt.

1 Het samenbrengen van tekst en PostScript grafisch materiaal

In dit artikel beschrijf ik enkele \LaTeX stijlen die, samen met Tom Rokicki's `dvips` [1] (en waarschijnlijk ook andere dvi-vertaalprogramma's) het mogelijk maken PostScript's uitgebreid opdrachtenpalet [2, 3, 4, 5, 6, 7] te gebruiken om bepaalde grafische effecten te verkrijgen. Het enige waar U voor moet opletten indien U gebruik wilt maken van deze functies, is dat hun effect slechts zichtbaar wordt indien U ook beschikt over een PostScript drukker of visualizator, een programma dat PostScript kan omzetten in een beeld op het scherm van een werkstation of een persoonlijke computer (PC of Mac). Een voorbeeld van zo'n programma in het publieke domein is `ghostview` (dat deel uitmaakt van het GNU software project).

Niettegenstaande het feit dat het mogelijk is om via de \TeX opdracht `\special` willekeurige teksten door te

spelen aan de dvi-vertaalprogramma's (in ons geval elementaire PostScript opdrachten), is het in het algemeen aan te raden meer generische constructies te gebruiken. De eerste stijl, die ik bespreek, heet `epsfig` en maakt het U eenvoudig om een PostScript beeld in te voeren in een L^AT_EX bestand. Zoals de naam het aanduidt gaat het hier over een 'Encapsulated PostScript' (EPS) [8] bestand d.w.z. een bestand dat, o.a. informatie bevat over zijn dimensies, meer in het bijzonder een regel bevat met de structuur:

```
%%BoundingBox: llx lly urx ury
```

met `llx` en `lly` de x- en y-dimensies (in PostScript punten, waar 1 PostScript punt = 0,35278mm) van de linker beneden hoek en `urx` en `ury` de x- en y-dimensies van de rechter bovenhoek van een (virtuele) rechthoek, die het gehele beeld omsluit. Aan de hand van deze dimensies kan de opdracht `epsfig` de plaats berekenen, die opengelaten moet worden om het beeld te kunnen afdrukken. Kleine beeldjes, zoals '⊗' en '∅' kunnen om het even overal in het document worden gebruikt.

De volledige syntax van de opdracht is:

```
\epsfig{file=fn,%
        height=ht,width=wd,%
        clip=,rotate=angle,silent,%
        bblx=blx,bbly=bly,%
        bburx=brx,bbury=bry}
```

<code>file</code>	De naam van het Encapsulated PostScript bestand (je kan ook <code>figure=</code> gebruiken).
<code>height</code>	De gewenste hoogte van het beeld op het gedrukt blad (in één van de T _E X eenheden). Indien deze parameter en de volgende (<code>width</code>) niet gespecificeerd zijn dan zal het beeld zijn 'natuurlijke' grootte hebben, d.w.z. de dimensies gespecificeerd op de <code>BoundingBox</code> lijn in het Encapsulated PostScript bestand zelf. Wanneer men de breedte (<code>width=</code>) specificiert en geen hoogte (<code>height=</code>), dan wordt deze laatste automatisch aangepast zodanig dat de verhouding tussen beide dimensies onveranderd blijft.
<code>width</code>	De gewenste breedte van het beeld op het gedrukt blad (in één van de T _E X eenheden). Indien deze parameter en de vorige (<code>height</code>) niet gespecificeerd zijn dan zal het beeld zijn 'natuurlijke' grootte hebben, d.w.z. de dimensies gespecificeerd op de <code>BoundingBox</code> lijn in het Encapsulated PostScript bestand zelf. Wanneer men de hoogte (<code>height=</code>) specificiert en geen breedte (<code>width=</code>), dan wordt deze laatste automatisch aangepast zodanig dat de verhouding tussen beide dimensies onveranderd blijft.
<code>clip</code>	Wanneer deze optie wordt gekozen, dan worden de delen van het beeld die buiten de

	'BoundingBox' vallen, weggeknipt. Opgelet dat " <code>clip=</code> " geen waarde neemt, maar dat het '='-teken aanwezig moet zijn.
<code>rotate</code>	De draaihoek (in graden, de positieve richting gaat tegen de richting van de wijzers van een uurwerk in) voor het gehele beeld.
<code>silent</code>	Met deze optie werkt <code>\epsfig</code> 'stille-tjes', d.w.z. zonder informatie op het beeldscherm (of in de log) te schrijven.
<code>bblx</code>	De x-coördinaat van de linker benedenhoek van de omschreven rechthoek (BoundingBox).
<code>bbly</code>	De y-coördinaat van de linker benedenhoek van de omschreven rechthoek (BoundingBox).
<code>bburx</code>	De x-coördinaat van de rechter bovenhoek van de omschreven rechthoek (BoundingBox).
<code>bbury</code>	De y-coördinaat van de rechter bovenhoek van de omschreven rechthoek (BoundingBox).

Wanneer de `BoundingBox` parameters **vóór** de naam van de figuur staan (`figure=` or `file=`), dan negeert `\epsfig` de parameters in het Encapsulated PostScript bestand (`\epsfig` zal dan zelfs het bewuste bestand niet trachten te lezen), en gebruikt de dimensies gespecificeerd in de opdracht zelf. Dat kan interessant zijn om een bepaald gedeelte van het beeld weg te knippen (clipping), of om PostScript bestanden te gebruiken die de `BoundingBox` informatie niet bevatten of waar deze laatste verkeerd is. U moet wel oppassen dat u geen spaties laat in de argumenten van `\epsfig`, want dan werkt die opdracht niet; als U verplicht bent een `\epsfig` opdracht over verschillende regels uit te schrijven, dan moet U elke lijn afsluiten met een %-teken (zoals ik ook deed in de beschrijving van de opdracht hierboven).

1.1 Simpele figuren

De eenvoudigste situatie is wanneer U tevreden bent met de 'natuurlijke' dimensies van het beeld (zoals gespecificeerd in het bestand zelf op de `BoundingBox` lijn). Dan heeft U geen schaalfactor nodig en `\epsfig` berekent de plaats die het plaatje inneemt eenvoudig uitgaande van deze dimensies. Een eenvoudige manier om de natuurlijke dimensies van een beeld te kennen is het te drukken op een PostScript printer. De linker benedenhoek van het beeld wordt gepositioneerd daar waar de `\epsfig` opdracht wordt gegeven. Indien de opgegeven breedte (`width`) of hoogte (`height`) niet overeenstemmen met de respectieve natuurlijke dimensies, dan wordt het beeld met de nodige schaalfactor vergroot of verkleind.

Figuur 1 op pagina 104 is een prent waarvoor we een hoogte van 5cm wensen. De `\epsfig` opdracht werd in een `\mbox` gestopt om de figuur te centreren met behulp van een `center` omgeving.

1.2 Proeffiguren

Sommige PostScript figuren hebben een lange tijd nodig om gedrukt te worden. Voor zulke figuren is er een ‘proef’ (draft) optie, die niet het PostScript beeld zelf afdrukt, maar ze vervangt door een rechthoek met dezelfde dimensies als het uiteindelijke beeld in de tekst zelf, tezamen met de naam van de figuur (zie figuur 2 op pagina 104). De opdracht `\psdraft` schakelt om naar proefmode, zodat alle volgende `epsfig` opdrachten ‘draft’ rechthoekjes, in plaats van de oorspronkelijke figuren, zullen drukken. De opdracht `\psfull`, keert terug naar de normale mode. Een ander voordeel van de draft-mode is dat er geen T_EX `\special` opdrachten in het dvi-bestand worden geschreven, zodat het test document bekeken kan worden met een willekeurig visualisatie-programma, dat niet noodzakelijk PostScript behoeft te verstaan.

1.3 Schikking van beeldmateriaal

We kunnen nu de mogelijkheden van de `\epsfig` opdracht en van L^AT_EX combineren om ons beeldmateriaal optimaal op het blad te schikken. De figuren 3 tot 5 op pagina 119 tonen schikkingen, verkregen door gebruik te maken van `minipage` omgevingen.

2 Het roteren van tekst- en beeldmateriaal

Het komt vaak voor dat we een figuur of tabel willen draaien, enerzijds om de presentatie van de informatie te verbeteren (bijvoorbeeld een tabel wijder dan de kolom breedte), anderzijds om bepaalde visuele effecten te bereiken. Een handige manier om deze draaifunctionaliteit te verkrijgen is met de stijl `rotating`, ontwikkeld door Sebastian Rahtz en Leonor Barroca [9], die enkele nieuwe L^AT_EX omgevingen definiëert die ik hierna één voor één onder de loep wil nemen.

2.1 Het roteren van tekst

De `rotate` omgeving draait het omsloten materiaal over het opgegeven aantal graden (positief is met de wijzers mee, zoals in PostScript). Deze omgeving neemt echter aan dat het materiaal geen plaats inneemt, zodat U deze omgeving kunt gebruiken om speciale effecten te verkrijgen. Plaats het T_EX commentaar teken `%` aan het einde van de opdracht `\begin{rotate}{52}%` — U moet de opdracht altijd volgen met tekst of met `%`, om te voorkomen dat ongewenste spaties in de tekst opduiken.

```
Begin van de zin \begin{rotate}{-52}%
NTG
\end{rotate} Einde van de zin
```

Begin van de zin ^{NTG} Einde van de zin

Indien U wenst dat L^AT_EX de nodige ruimte openlaat om het materiaal te plaatsen, dan gebruikt U de omgeving `turn`:

```
Begin van de zin \begin{turn}{52}%
NTG
\end{turn} Einde van de zin
```

Begin van de zin ^{NTG} Einde van de zin

De omgeving ‘`sideways`’ is een speciaal geval, van de `turn` omgeving, met een draaihoek van -90° :

```
Begin van de zin \begin{sideways}%
NTG
\end{sideways} Einde van de zin
```

Begin van de zin ^{NTG} Einde van de zin

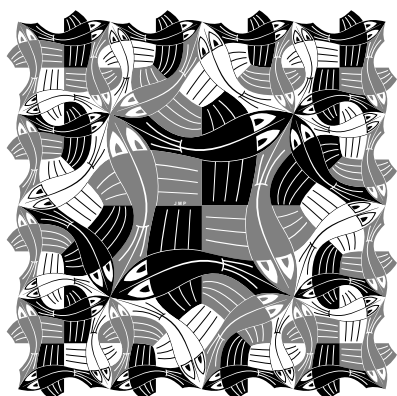
Wanneer men met hele paragrafen tekst werkt, dan vindt men dat T_EX ramen (‘boxes’) in het algemeen zowel een hoogte als een diepte hebben. In dit geval vinden de rotaties plaats rond het uiterste linker punt waar het raam de basislijn raakt. Dit kan soms tot eigenaardige en onverwachte resultaten leiden, zoals figuur 6 op pagina 120 toont. Voor elk beeldje tonen de lijntjes links en rechts de positie van de basislijn. U kunt de positionering van de ramen min of meer controleren door het specificeren van de (optionele) positie parameter die beschikbaar zijn bij bepaalde L^AT_EX opdrachten.

```
\newcommand{\B}{A b C d E f G h I j K
l M n O p Q}
Begin \begin{turn}{45}
\parbox[t]{16mm}{\B}
\end{turn}
Midden \begin{turn}{45}
\parbox[b]{16mm}{\B}
\end{turn}
Einde
```

Begin ^{AbCdEf} ^{GhIjKl} ^{MnOpQ} Midden ^{AbCdEf} ^{GhIjKl} ^{MnOpQ} Einde

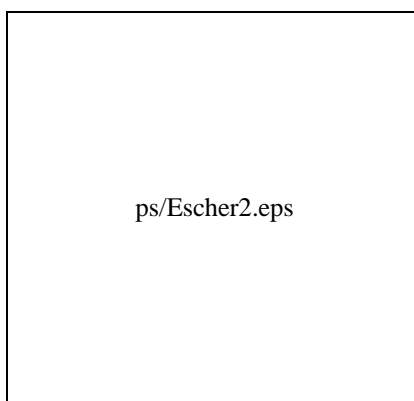
2.2 Het roteren van tabellen

Materiaal in tabellen (L^AT_EX’s `tabular` en `array` omgevingen) kunnen op een analoge manier gedraaid worden. De voorbeelden in figuur 7 op pagina 120 tonen hoe ‘onzichtbare regels’ gebruikt worden om de hoogte of de breedte van de kolommen te controleren.



```
\begin{center}
  \mbox{\epsfig{file=ps/Escher2.eps,height=50mm}}
\end{center}
\caption[] {Een eenvoudig gecentreerde figuur}
\label{fig:simple}
```

Figuur 1: Een eenvoudig gecentreerde figuur



```
% ga naar test (draft) mode
\psdraft

\begin{center}
  \mbox{\epsfig{file=ps/Escher2.eps,height=50mm}}
\end{center}
\caption[] {Een figuur in test (draft) mode}
\label{fig:draft}
% ga terug naar de normale mode
\psfull
```

Figuur 2: Een figuur in test (draft) mode

Rotaties kunnen gekombineerd worden, als volgt:

```
\begin{sideways}
\begin{tabular}{l@{\quad}r}
\em Zin \rule{0pt}{3cm}
      & \begin{rotate}{-90}%
Frequentie\end{rotate} \\
\hline
Goeie dag & 33\\
Tot ziens & 34\\
\hline
\end{tabular}
\end{sideways}
```

Frequentie	33	34
Zin	Goeie dag	Tot ziens

Een ingewikkelder (en wellicht meer voorkomend) voorbeeld is tabel 1 op pagina 105, die een sideways omgeving in een sidewaysstable omgeving toont. Deze laatste omgeving draait niet enkel de tabel zelf, maar ook zijn onderschrift. De sideways-table omgeving — en zijn analoog voor figuren, de sidewaysfigure omgeving — nemen echter een volledig pagina in, en men moet dus wel een beetje uitkijken om zeker te zijn dat men de pagina wel degelijk op een aanvaardbare manier kan vullen.

2.3 Het roteren van figuren

Met de `rotating` stijl kunt U niet enkel tekst maar ook PostScript beeldmateriaal draaien. Een voorbeeld ziet U in figuur 8 op pagina 121, waar een Encapsulated PostScript beeld in het dokument wordt ingevoerd met de `\epsfig` opdracht, en daarna wordt gedraaid naar keuze. Het is interessant om voor elk geval naar de positie van de basislijn te kijken.

3 Het kleuren van ramen

Om de nadruk te leggen op een bepaald gedeelte van de tekst is het dikwijls nuttig het bedoelde materiaal te omlijnen of het met een grijze achtergrond te kleuren.

De stijl `psboxit` (geschreven door J. Maillot) bouwt een PostScript raam boven op een T_EX raam, gebruik makend van de parameters berekend voor deze laatste. Om de mogelijkheden van deze stijl te benutten moeten de PostScript opdrachten geïnitieerd worden, door de opdracht `\PScommands` te geven aan het begin van het dokument.

De basisopdracht is `\psboxit`, gedefinieerd als volgt:

```
\psboxit{PS opdrachten}{TEX materiaal}
```

Deze opdracht onderwerpt het materiaal gespecificeerd in het tweede argument ‘T_EX materiaal’ aan het effect van de PostScript opdrachten in het eerste argument ‘PS opdrachten’. Het stijlbestand definiëert verscheidene PostScript procedures, zoals `cartouche` en `rectcartouche`. Deze laten toe zekere visuele effecten te verkrijgen, zoals getoond in het voorbeeld hieronder:

```
--\psboxit{5 cartouche}{NTG}--
--\psboxit{rectcartouche}{\spbox{DANTE}}--
-
--\psboxit{box .7 setgray fill}%
    {\spbox{TUG}}--
```

```
NTG-- DANTE -- TUG --
```

De bijkomende opdracht `\spbox` werkt zoals `\fbox`, d.w.z. men verkrijgt een raam rond het ingeschreven materiaal met een afstand `\fboxsep` toegevoegd in elke richting. Het enige verschil is dat `\spbox` geen lijnen afbeeldt rond het raam. Het belangrijkste doel van `\fboxsep` is raam te bouwen met identieke afmetingen als `\fbox`, maar met een gekleurde achtergrond. In het algemeen kunt U natuurlijk zelf opdrachten definiëren, zoals:

```
\newcommand{\grijsraam}[1]%
{\psboxit{box .7 setgray fill}{\fbox{#1}}}
```

Indien U grotere ramen wilt kleuren, dan gebruikt U het best de `boxitpara` omgeving:

```
\begin{boxitpara}{box 0.7 setgray fill}
Amsterdam is een grote stad.
Antwerpen heeft een mooie kathedraal.
Groningen ligt in het noorden.
Leuven ligt in het zuiden.
\end{boxitpara}
```

```
Amsterdam is een grote stad. Antwerpen heeft een
mooie kathedraal. Groningen ligt in het noorden.
Leuven ligt in het zuiden.
```

3.1 Werken met kleuren

Echte kleuren, in tegenstelling met verschillende grijsniveau's, zijn natuurlijk enkel te verkrijgen wanneer men over een kleurendrukker beschikt. `dvips` maakt het mogelijk om PostScript's kleuropdrachten te gebruiken, en stelt hiervoor de stijl `dpcolor` en `dpblack` ter beschikking. Deze laatste stijl neutraliseert alle PostScript kleuropdrachten van `dpcolor`, zodat men een bestand dat deze opdrachten bevat, zonder problemen kan afdrukken op een zwart-wit printer. `dpcolor` biedt een pallet van 68 kleuren (de complete lijst vindt U in het stijlbestand zelf).

De kleur van de achtergrond voor de lopende en volgende bladzijden wordt bepaald door de opdracht `\background`, met als parameter de naam van de kleur. Indien men een groene achtergrond wenst dan zegt men:

```
\background{Green}
```

Wanneer men dan terug wilt naar de gewone (default) witte achtergrond, dan geeft men:

```
\background{White}
```

`dpcolor` heeft twee soorten kleuropdrachten; de eerste heeft één argument, om korte teksten te kleuren, bijvoorbeeld:

```
\Blue{De volgende tekst wordt in 't
blauw gezet.}
```

De volgende tekst wordt in 't blauw gezet.

terwijl de tweede de kleur voor de tekst herdefiniëert, en deze definitie onveranderd laat tot aan de volgende kleuropdracht.

```
\textGreen De volgende tekst is in
't groen. En dat bl"yft ie tot we
de kleur herdefini"eren.
\textBlack De tekst wordt terug zwart
```

De volgende tekst is in 't groen. En dat blijft ie tot we de kleur herdefiniëren. De tekst wordt terug zwart

Bijkomende kleuren kunnen gedefinieerd worden in functie van het CMYK (Cyan, Magenta, Yellow, black) kleurenmodel, waarin men een kleur specificeert aan de hand van de intensiteit (een getal tussen 0.0 en 1.0) van elk van haar vier kleurencomponenten.

```
\Color{.2 .3 .4 .5}{De kleur van deze
tekst werd gespecifi"eerd als
een CMYK kwadruplet}
```

De kleur van deze tekst werd gespecificeerd als een CMYK kwadruplet

References

- [1] Tomas Rokicki. *DVIPS: A T_EX Driver*. dvips Distribution, 1993.
- [2] Adobe Systems Incorporated. *PostScript Language Reference Manual*. Addison-Wesley, Reading, Second edition, 1990.
- [3] Adobe Systems Incorporated. *PostScript Language Tutorial and Cookbook*. Addison-Wesley, Reading, 1985.
- [4] Glenn C. Reid. *PostScript Language Program Design*. Addison-Wesley, Reading, 1988.
- [5] Glenn C. Reid. *Thinking in PostScript*. Addison-Wesley, Reading, 1990.
- [6] Stephen E. Roth, editor. *Real World PostScript*. Addison-Wesley, Reading, 1988.
- [7] Ross Smith. *Learning PostScript—A Visual Approach*. Peachpit Press, 1085 Keith Avenue, Berkeley, CA 94708, 1990.
- [8] Peter Vollenmeider. *Encapsulated PostScript: Applications for the Macintosh and PC*. Prentice-Hall and Carl Hanser, 1990.
- [9] Sebastian Rahtz and Leonor Barroca. A style option for rotated objects in L^AT_EX. *TUGBoat*, 13(2):156–180, July 1992.
- [10] Karl Berry. Filenames for fonts. *TUGBoat*, 11(4):517–520, November 1990.

Familie	Serie	Vorm	Externe namen
Families met schreef (serif)			
Times	m	n, it	Times-Roman(ptmr), Times-Italic(ptmri)
	b	n, it	Times-Bold(ptmb), Times-BoldItalic(ptmbi)
Palatino	m	n, it	Palatino-Roman(pplr), Palatino-Italic(pplr i)
	b	n, it	Palatino-Bold(pplb), Palatino-BoldItalic(pplbi)
NewCenturySchoolbook	m	n, it	NewCenturySchlbk-Roman(pncr), NewCenturySchlbk-Italic(pncri)
	b	n, it	NewCenturySchlbk-Bold(pncb), NewCenturySchlbk-BoldItalic(pncbi)
Bookman	m	n, it	Bookman-Light(pbkl), Bookman-LightItalic(pbkli)
	b	n, it	Bookman-Demi(pbkd), Bookman-DemiItalic(pbkdi)
Schreefloze (sans serif) families			
Helvetica	m	n, it	Helvetica(phvr), Helvetica-Oblique(phvro)
	b	n, it	Helvetica-Bold(phvb), Helvetica-BoldOblique(phvbo)
	c	n, it	Helvetica-Narrow(phvrrn), Helvetica-Narrow-Oblique(phvron)
	bc	n, it	Helvetica-Narrow-Bold(phvbrn), Helvetica-Narrow-BoldOblique(phvbon)
AvantGarde	m	n, it	AvantGarde-Book(pagk), AvantGarde-BookOblique(pagko)
	b	n, it	AvantGarde-Demi(pagd), AvantGarde-DemiOblique(pagdo)
Schrijfmachine font			
Courier	m	n, it	Courier(pcr), CourierOblique(pcrro)
	b	n, it	Courier-Bold(pcrb), Courier-BoldOblique(pcrbo)
Versier en fantasie fonts			
Symbol	m	n	Symbol(psyr)
ZapfChancery	m	n	ZapfChancery-MediumItalic(pzcmi)
ZapfDingbats	m	n	ZapfDingbats(pzdr)

Tabel 2: Klassificatie van de 35 basis PostScript fonts (tussen haakjes de naam volgens K. Berry)

	0	1	2	3	4	5	6	7	8	9
30										
40	→	↗	↘	↙	↘	↗	↖	↗	↘	↙
50	↖	↗	↘	↙	↘	↗	↖	↗	↘	↙
60	+	+	+	+	+	+	+	+	+	+
70	+	+	+	+	+	+	+	+	+	+
80	*	*	*	*	*	*	*	*	*	*
90	*	*	*	*	*	*	*	*	*	*
100	*	*	*	*	*	*	*	*	*	*
110	■	□	□	□	□	▲	▼	◆	◇	▼
120	—	—	—	—	—	—	—	—	—	—
160	—	—	—	—	—	—	—	—	—	—
170	♥	♠	♠	♠	♠	♠	♠	♠	♠	♠
180	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
190	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
200	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
210	⊗	⊗	→	→	→	→	→	→	→	→
220	→	→	→	→	→	→	→	→	→	→
230	→	→	→	→	→	→	→	→	→	→
240	→	→	→	→	→	→	→	→	→	→
250	→	→	→	→	→	→	→	→	→	→

Tabel 3: Het ZapfDingbats PostScript font

	0	1	2	3	4	5	6	7	8	9
30				!	∇	#	∃	%	&	∃
40	()	*	+	.	-	.	/	0	1
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	≡	A	B	X	Δ	E
70	Φ	Γ	H	P	I	Σ	∅	M	N	O
80	Π	Θ	P	Σ	T	K	Y	Ω	Ε	Ψ
90	Z	[:]	⊥	⊥	⊥	α	β	χ
100	δ	ε	φ	π	γ	η	ρ	κ	λ	μ
110	v	ε	o	ψ	θ	ι	σ	υ	ω	ω
120	ε	ψ	ζ	·	≤	/	∞	~	±	♦
160	—	—	—	—	—	—	—	—	—	—
170	▲	↔	↑	↑	↑	→	↓	±	±	♥
180	x	x	∂	•	+	+	≠	≡	≈	—
190	—	⊥	⊥	∩	∩	∩	∩	⊗	⊕	∩
200	∩	∩	∩	∩	∩	∩	∩	∩	∩	∩
210	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
220	←	←	←	←	←	←	←	←	←	←
230	←	←	←	←	←	←	←	←	←	←
240	←	←	←	←	←	←	←	←	←	←
250	←	←	←	←	←	←	←	←	←	←

Tabel 4: Het Symbol PostScript font

a	α	b	β	c	χ	d	δ	e	ε	f	φ	g	γ	h	η	i	ι	j	φ	k	κ	l	λ	m	μ
n	ν	o	ο	p	π	q	θ	r	ρ	s	σ	t	τ	u	υ	v	ϖ	w	ω	x	ξ	y	ψ	z	ζ
A	Α	B	Β	C	Χ	D	Δ	E	Ε	F	Φ	G	Γ	H	Η	I	Ι	J	Θ	K	Κ	L	Λ	M	Μ
N	Ν	O	Ο	P	Π	Q	Θ	R	Ρ	S	Σ	T	Τ	U	Υ	V	ς	W	Ω	X	Ξ	Y	Ψ	Z	Ζ

Tabel 5: De Griekse karakters in het Symbol PostScript font

References

- [1] Tomas Rokicki. *DVIPS: A T_EX Driver*. dvips Distribution, 1993.
- [2] Adobe Systems Incorporated. *PostScript Language Reference Manual*. Addison-Wesley, Reading, Second edition, 1990.
- [3] Adobe Systems Incorporated. *PostScript Language Tutorial and Cookbook*. Addison-Wesley, Reading, 1985.
- [4] Glenn C. Reid. *PostScript Language Program Design*. Addison-Wesley, Reading, 1988.
- [5] Glenn C. Reid. *Thinking in PostScript*. Addison-Wesley, Reading, 1990.
- [6] Stephen E. Roth, editor. *Real World PostScript*. Addison-Wesley, Reading, 1988.
- [7] Ross Smith. *Learning PostScript—A Visual Approach*. Peachpit Press, 1085 Keith Avenue, Berkeley, CA 94708, 1990.
- [8] Peter Vollenmeider. *Encapsulated PostScript: Applications for the Macintosh and PC*. Prentice-Hall and Carl Hanser, 1990.
- [9] Sebastian Rahtz and Leonor Barroca. A style option for rotated objects in L^AT_EX. *TUGBoat*, 13(2):156–180, July 1992.
- [10] Karl Berry. Filenames for fonts. *TUGBoat*, 11(4):517–520, November 1990.

Familie	Serie	Vorm	Externe namen
Families met schreef (serif)			
Times	m	n, it	Times-Roman(ptmr), Times-Italic(ptmri)
	b	n, it	Times-Bold(ptmb), Times-BoldItalic(ptmbi)
Palatino	m	n, it	Palatino-Roman(pplr), Palatino-Italic(pplr i)
	b	n, it	Palatino-Bold(pplb), Palatino-BoldItalic(pplbi)
NewCenturySchoolbook	m	n, it	NewCenturySchlbk-Roman(pncr), NewCenturySchlbk-Italic(pncri)
	b	n, it	NewCenturySchlbk-Bold(pncb), NewCenturySchlbk-BoldItalic(pncbi)
Bookman	m	n, it	Bookman-Light(pbkl), Bookman-LightItalic(pbkli)
	b	n, it	Bookman-Demi(pbkd), Bookman-DemiItalic(pbkdi)
Schreefloze (sans serif) families			
Helvetica	m	n, it	Helvetica(phvr), Helvetica-Oblique(phvro)
	b	n, it	Helvetica-Bold(phvb), Helvetica-BoldOblique(phvbo)
	c	n, it	Helvetica-Narrow(phvrrn), Helvetica-Narrow-Oblique(phvron)
	bc	n, it	Helvetica-Narrow-Bold(phvbrn), Helvetica-Narrow-BoldOblique(phvbon)
AvantGarde	m	n, it	AvantGarde-Book(pagk), AvantGarde-BookOblique(pagko)
	b	n, it	AvantGarde-Demi(pagd), AvantGarde-DemiOblique(pagdo)
Schrijfmachine font			
Courier	m	n, it	Courier(pcr), CourierOblique(pcrro)
	b	n, it	Courier-Bold(pcrb), Courier-BoldOblique(pcrbo)
Versier en fantasie fonts			
Symbol	m	n	Symbol(psyr)
ZapfChancery	m	n	ZapfChancery-MediumItalic(pzcmi)
ZapfDingbats	m	n	ZapfDingbats(pzdr)

Tabel 2: Klassificatie van de 35 basis PostScript fonts (tussen haakjes de naam volgens K. Berry)

	0	1	2	3	4	5	6	7	8	9
30										
40	+	+	+	+	+	+	+	+	+	+
50	+	+	+	+	+	+	+	+	+	+
60	+	+	+	+	+	+	+	+	+	+
70	+	+	+	+	+	+	+	+	+	+
80	+	+	+	+	+	+	+	+	+	+
90	+	+	+	+	+	+	+	+	+	+
100	+	+	+	+	+	+	+	+	+	+
110	+	+	+	+	+	+	+	+	+	+
120	+	+	+	+	+	+	+	+	+	+
160	+	+	+	+	+	+	+	+	+	+
170	+	+	+	+	+	+	+	+	+	+
180	+	+	+	+	+	+	+	+	+	+
190	+	+	+	+	+	+	+	+	+	+
200	+	+	+	+	+	+	+	+	+	+
210	+	+	+	+	+	+	+	+	+	+
220	+	+	+	+	+	+	+	+	+	+
230	+	+	+	+	+	+	+	+	+	+
240	+	+	+	+	+	+	+	+	+	+
250	+	+	+	+	+	+	+	+	+	+

Tabel 3: Het ZapfDingbats PostScript font

	0	1	2	3	4	5	6	7	8	9
30										
40	()	*	+	.	-	.	/	&	1
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	≡	A	B	X	Δ	E
70	Φ	Γ	H	P	I	Σ	θ	T	N	O
80	Π	Θ	P	Σ	I	Σ	θ	T	N	O
90	Z	[:]	⊥	⊥	⊥	⊥	α	β
100	δ	ε	φ	π	γ	η	ρ	ι	σ	κ
110	v	v	v	v	v	v	v	v	v	v
120	ε	ψ	γ	ζ	ζ	ζ	ζ	ζ	ζ	ζ
160	ε	ψ	γ	ζ	ζ	ζ	ζ	ζ	ζ	ζ
170	▲	↔	↔	↔	↔	↔	↔	↔	↔	↔
180	x	x	x	x	x	x	x	x	x	x
190	—	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
200	⊂	⊃	⊄	⊅	⊆	⊇	⊈	⊉	⊊	⊋
210	⊌	⊍	⊎	⊏	⊐	⊑	⊒	⊓	⊔	⊕
220	⊖	⊗	⊘	⊙	⊚	⊛	⊜	⊝	⊞	⊟
230	(
240										
250										

Tabel 4: Het Symbol PostScript font

a	α	b	β	c	χ	d	δ	e	ε	f	φ	g	γ	h	η	i	ι	j	φ	k	κ	l	λ	m	μ
n	v	o	o	p	π	q	θ	r	ρ	s	σ	t	τ	u	υ	v	ω	w	ω	x	ξ	y	ψ	z	ζ
A	A	B	B	C	X	D	Δ	E	E	F	Φ	G	Γ	H	H	I	Ι	J	Θ	K	K	L	Λ	M	M
N	N	O	O	P	Π	Q	Θ	R	P	S	Σ	T	T	U	Υ	V	ς	W	Ω	X	Ξ	Y	Ψ	Z	Z

Tabel 5: De Griekse karakters in het Symbol PostScript font

```

\begin{figure*}[p]
  \begin{minipage}[b]{.495\linewidth}
    \centering\mbox{\epsfig{figure=ps/Scandinavia.eps,width=.9\textwidth}}
    \caption{Noord Europa} \label{fig:draai1}
  \end{minipage}\hfill
  \begin{minipage}[b]{.495\linewidth}
    \centering\mbox{\epsfig{figure=ps/Africa.eps,width=.9\textwidth}}
    \caption{De kaart van Afrika} \label{fig:draai2}
  \end{minipage}
  \centering\mbox{\epsfig{figure=ps/USA.eps,width=.70\textwidth}}
  \caption{De Verenigde Staten van Amerika} \label{fig:draai3}
\end{figure*}

```



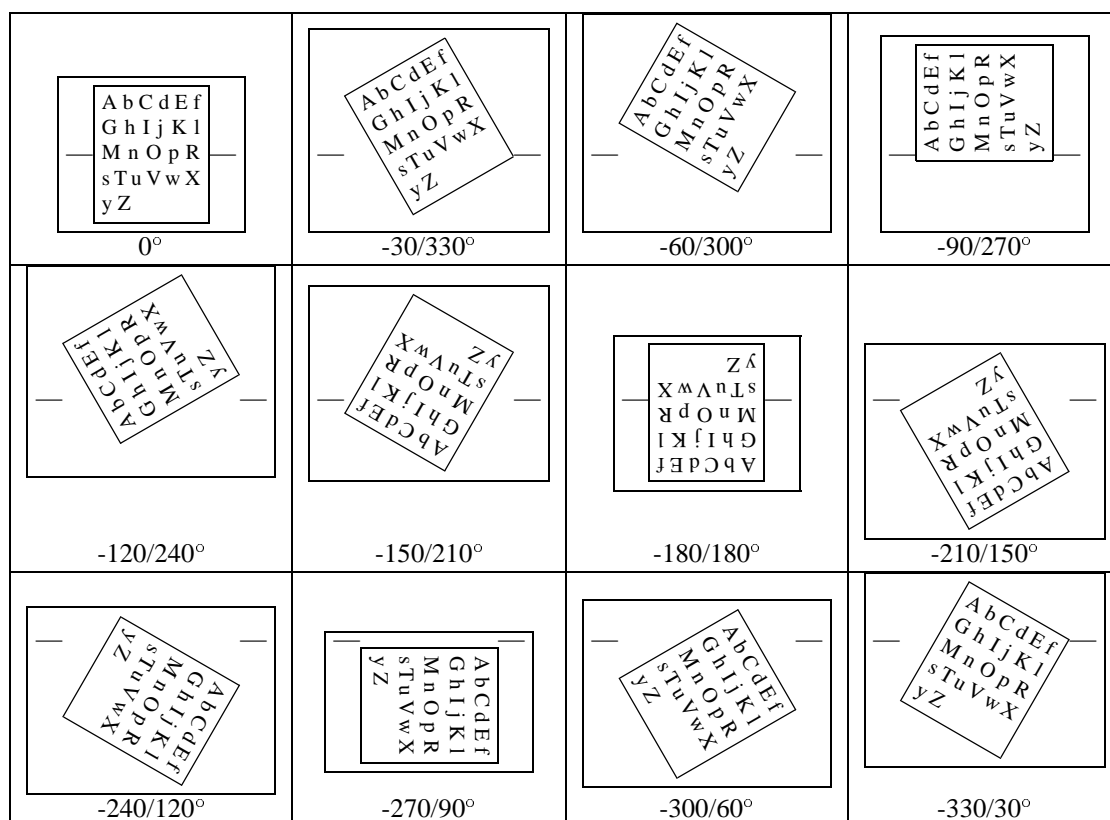
Figuur 3: Noord Europa



Figuur 4: De kaart van Afrika



Figuur 5: De Verenigde Staten van Amerika



Figuur 6: Het draaien van paragrafen

Kolom 1	Kolom 2	Kolom 3
A	B	C
D	E	F
G	H	I

```
\begin{tabular}{rrr}\l[3mm]
\begin{rotate}{-45}Kolom 1\end{rotate}&
\begin{rotate}{-45}Kolom 2\end{rotate}&
\begin{rotate}{-45}Kolom 3\end{rotate}\l
\hline A& B& C\l D& E& F\l G& H& I\l \hline
\end{tabular}
```

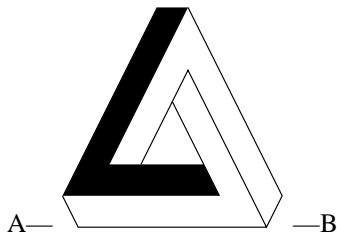
Kolom 1	Kolom 2	Kolom 3
A	B	C
D	E	F
G	H	I

```
\begin{tabular}{rrr}
\begin{turn}{-45}Kolom 1\end{turn}&
\begin{turn}{-45}Kolom 2\end{turn}&
\begin{turn}{-45}Kolom 3\end{turn}\l
\hline A& B& C\l D& E& F\l G& H& I\l\l\hline
\end{tabular}
```

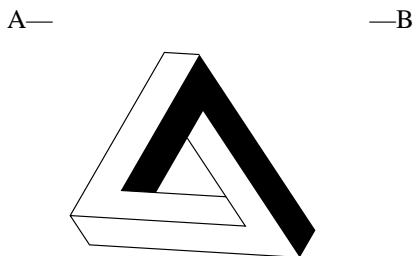
Kolom 1	Kolom 2	Kolom 3
A	B	C
D	E	F
G	H	I

```
\begin{tabular}{rrr}\l[5mm]
\begin{rotate}{-45}Kolom 1\end{rotate}
\rule{.5cm}{0pt}&
\begin{rotate}{-45}Kolom 2\end{rotate}
\rule{.5cm}{0pt}&
\begin{rotate}{-45}Kolom 3\end{rotate}
\rule{.5cm}{0pt}\l
\hline A& B& C\l D& E& F\l G& H& I\l\l\hline
\end{tabular}
```

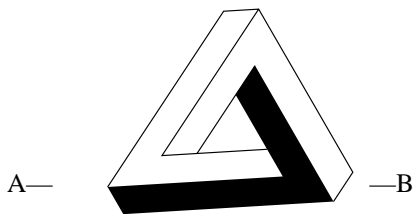
Figuur 7: Het draaien van informatie in tabellen



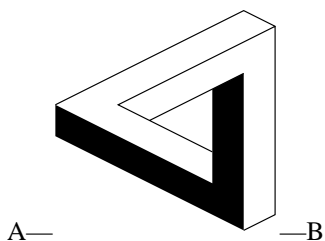
```
A---
  \epsfig{figure=ps/Escher3.eps,width=30mm}
---B
```



```
A---\begin{turn}{-240}
  \epsfig{figure=ps/Escher3.eps,width=30mm}
\end{turn}---B
```



```
A---\begin{turn}{240}
  \epsfig{figure=ps/Escher3.eps,width=30mm}
\end{turn}---B
```



```
A---\begin{sideways}
  \epsfig{figure=ps/Escher3.eps,width=30mm}
\end{sideways}---B
```

Figuur 8: *Verskillende manieren om een beeld te draaien*