

Michael van Hartskamp

Venuslaan 405, 5632 HM Eindhoven

michael@vanhartskamp.com

T_EX-rubriek

PDFTEX II

In de vorige aflevering besprak Michael van Hartskamp, gepromoveerd topoloog, het Portable Document Format (PDF), dat het mogelijk maakt om teksten onafhankelijk van het computersysteem te zetten. Dit programma blijkt ook geschikt te zijn voor het maken van transparanten voor een presentatie. Hoe dat in zijn werk gaat, wordt deze keer besproken.

Meer en meer wiskundigen laten hun krijtje liggen bij het geven van een presentatie. Dit doen ze vaak uitsluitend om deze te vervangen door handgeschreven slides, of het andere uiterste: presenteren met het Windows-programma Powerpoint. Voor een wiskundige is het van het grootste belang dat de formules correct op het scherm staan en Powerpoint is daarbij eerder last dan steun. Anderzijds, wie veel met niet-wiskundigen (bijvoorbeeld in bedrijven) samenwerkt kan eigenlijk niet meer met ouderwetse transparanten aankomen.

Er is gelukkig nog een mogelijkheid: slides maken met (pdf)L^AT_EX. In deze aflevering bekijken we de mogelijkheden. Gelukkig zijn dat er meer dan menigeen zou denken.

Maar eerst een stapje terug in het verleden. Wie ooit met SliT_EX slides heeft gemaakt kan het zich ongetwijfeld herinneren: het was een en al gepruts. SliT_EX was dan ook een variant op L^AT_EX en de relatie tussen de twee was nogal merkwaardig. SliT_EX had daarboven nog extra grote fonts nodig en vroeg bij gebruik nogal veel van de toenmalige PC's.

De 'opvolgers' van SliT_EX zoals de `slides` class waren wat dat betreft aanmerkelijk prettiger in de omgang. Een nadeel was dat ze veelal gebaseerd waren op onderliggende eigenschappen van postscriptprinters. Dit maakte dat het resultaat op papier of transparant prima was, mits je over een postscriptprinter beschikte, maar op het scherm bekijken met ghostscript/view was géén optie. Dit was niet alleen lelijk, ook lang niet op alle computers was ghostscript/view geïnstalleerd en dat bleek toch een voorwaarde voor het succesvol presenteren buiten de deur zonder meteen een eigen laptop mee te moeten nemen.

De vorige keer bekeken we pdfT_EX en Acrobat. Dankzij de beschikbaarheid van de Acrobat reader op de meeste computerplatforms (Windows, Unix, ...) en de goede verspreidingsgraad is met pdfT_EX presentaties maken en met Acrobat vertonen een levensvatbaar alternatief.

De eerder genoemde op postscript gebaseerde oplossingen werken dan echter veelal niet. Bovendien, het pdf-formaat biedt méér mogelijkheden dan postscript op het gebied van effecten en interactie. Voor meer informatie over pdfT_EX en hoe het te gebruiken, zie de aflevering in het nummer van december 2002.

In deze aflevering richten we ons specifiek op de mogelijkheid om met pdfT_EX presentaties te maken, voor gebruik met een beamer, en we doen dat in het bijzonder met het L^AT_EX-package `pdfscreen`.

Naast `pdfscreen` zijn er nog meer mogelijkheden. Een nog niet afgerond overzicht is te vinden op de website <http://www.miwie.org/presentations>. Hier staan ook referenties naar andere L^AT_EX-packages met een vergelijkbare doelstelling.

Als er eenmaal een pdf-bestand met een presentatie daarin is gemaakt, is presenteren vervolgens een fluitje van een cent. Wel verdient het aanbeveling om de gastheer of -vrouw te vragen of Acrobat reader beschikbaar is. Bij de presentatie zelf is de full-screen optie onmisbaar, hiermee is alleen de presentatie in beeld en zijn storende vensterranden, scrollbalken en dergelijke uit beeld. Meestal is dit effect te bereiken via de toetscombinatie `CTRL+L` en is het ongedaan maken gekoppeld aan de `ESC`-toets.

Het maken van de presentatie vereist iets meer technische vaardigheden, maar uiteindelijk valt het allemaal reuze mee. Het pakket `pdfscreen` is een opvolger van `pdfslide` en nu helemaal gericht op alle documenten op het beeldscherm. Het is daarmee bijvoorbeeld geschikt voor on-line dictaten, wat te danken is aan de opgenomen inhoudsopgave en de navigatieknoppen in het zogenaamde panel, waarover later meer. Een nadeel is dat 'gewone' slides maken iets lastiger is. Daar staat tegenover dat het heel eenvoudig is dezelfde tekst tot een elektronisch document en een papieren versie te bewerken. Probeer vooral de verleiding te weerstaan om nu een artikel in zijn geheel op slide te zetten. Laten we een voorbeeld geven.

```
\documentclass[a4paper]{article}
...
\usepackage[screen]{pdfscreen}
...
\begin{document}
```

Op de [a4paper] in de eerste regel komen we later terug. De optie [screen] geeft aan dat we een schermpresentatie maken. Het alternatief is [print]. De printversie levert een ‘gewoon’ document op, nagenoeg zonder de extra’s van pdfscreen. We zullen dan pdfTeX voor beide versies een keer moeten uitvoeren. Met behulp van

```
\begin{screen}... \end{screen}
```

respectievelijk

```
\begin{print}... \end{print}
```

is het mogelijk een fijner onderscheid te maken tussen de twee versies. De ingesloten tekst en TeX-codes worden namelijk alleen in de schermrespectievelijk printversie opgenomen.

Dit geldt overigens niet alleen ná maar ook vóór \begin{document} en kan zo worden gebruikt voor het doen van bijzondere instellingen voor elk.

De plaats van \usepackage{pdfscreen} moet ook nu weer zorgvuldig gekozen worden. Het pakket pdfscreen past namelijk een aantal bestaande TeX-commando’s aan en is daarom alleen werkzaam indien deze commando’s daarna niet weer gewijzigd worden. Dit is een van de vervelende problemen waar de TeX-gebruiker regelmatig geconfronteerd mee wordt. Het is niet anders.

Laat ons terugkeren naar de afmetingen. De optie [a4paper] is uitsluitend van toepassing op de te printen versie. Voor de schermversie zijn er de afzonderlijke opdrachten

```
\screensize{6.25in}{8in}
\margins{.75in}{.75in}{.75in}{.75in}
```

De gegeven waarden zijn de door de maker van pdfscreen aanbevolen afmetingen. Eerste de hoogte, daarna volgt de breedte en achtereenvolgens links, rechts, boven en onder voor margins.

De marges spreken voor zich. Wat betreft screensize, dit geeft in de eerste plaats een verhouding breedte/lengthe die vrij dicht tegen de klassieke 4/3 verhoudingen van tv’s en monitoren aan ligt (breedbeeld is 16/9). Het is belangrijk deze 4/3 verhouding te respecteren voor optimaal gebruik van het beschikbare geprojecteerde beeld. (Voor overhead transparanten of presentaties op breedbeeld plasmaschermen zijn andere waarden gunstiger.)

De absolute grootte bepaalt echter hoeveel er op een schermpagina komt. Na 6.25 inches is de bladzijde ‘vol’ en wordt automatisch aan een nieuwe begonnen. Het opvullen van een pagina gaat met tekens van ‘normale’ TeX-afmetingen. Naar mijn smaak staat er voor een presentatie te veel tekst op een pagina en is een hoogte van 4.2–4.5 inches beter.

Het pakket pdfscreen kent ook een slide-omgeving. Binnen

```
\begin{slide}... \end{slide}
```

zijn vanzelf al grotere lettertypen van kracht. Ook de vormgeving is iets anders dan er buiten. In de printversie worden deze slides omkaderd en ernaast is ruimte gereserveerd voor aantekening. Dit is ideaal voor hand-outs.

Eén van de bijzonderheden van pdfscreen is het zogenaamde panel. Het panel bevat een logo, de inhoudsopgave, en enkele navigatieknoppen. Die knoppen zijn eigenlijk vooral zinvol voor elektronische dictaten en niet zo heel erg voor presentaties.

Door gebruikmaking van \emblem{logo.jpg} kan een logo worden geïmporteerd uit een jpeg-bestand. Op iedere slide verschijnt nu automatisch het (instituu)logo. Dit draagt bij aan de herkenbaarheid van de presentatie en is een echte aanrader. Vooral het gemak van

inclusie en plaatsing is een groot voordeel van pdfscreen.

De inhoudsopgave wordt alleen opgenomen in het panel als paneltoc bij de opties in de \usepackage-regel staat vermeld. Met name bij langere inhoudsopgaven wordt opname soms erg lelijk.

Het is ook mogelijk de plaats van het panel te kiezen. panelleft en panelright zijn de opties, met de te verwachten resultaten.

Voor een ‘eenvoudige’ presentatie is nopanel te prefereren. Wie dan toch nog knoppen wil kan deze bijvoorbeeld met \bottombuttons in het klein opnemen onderaan iedere slide.

Een andere manier om een geheel eigen presentatie te maken is het gebruik van \overlay. Hiermee kan een plaatje op de achtergrond worden gezet. Dit kan natuurlijk ook weer een logo bevatten, of een foto van de campus of de spreker. De mogelijkheden zijn onbeperkt. Wie dit te afleidend vindt, kan volstaan met \backgroundcolor. Hiermee kan een egaal kleurtje worden ingesteld. Het spreekt voor zich dat het kleurenpakket color nog steeds effectief is binnen pdfLaTeX/pdfscreen. Veel kleurencombinaties zijn overigens al voorgeprogrammeerd.

Dat brengt ons tot het laatste onderwerp, dat bij menig presentator helaas soms tot hoofddoel is verheven: speciale effecten. Zelf ben ik absoluut geen liefhebber van te veel vliegende pagina’s, openende gordijnen, enz. We zullen hier een deel van de mogelijkheden schetsen.

Het standaard effect in pdfscreen om van pagina te verwisselen is het zogenaamde /R effect van replace en betekent: geen effect. Er zijn een aantal andere effecten beschikbaar via \pagedissolve. Dit zijn bijvoorbeeld: /Split dat op het openen van gordijnen lijkt en /Box waarin scheiding tussen de nieuwe en oude pagina door een steeds groter wordende rechthoek wordt aangegeven. Voor de effecten kan een totale tijdsduur worden aangegeven en voor sommige ook een richting.

Het pakket pdfscreen ondersteunt geen effect binnen een pagina. Ook daar is inmiddels een oplossing voor: PPower4 is een speciale postprocessor voor pdf-bestanden. Dit in Java geschreven programma bewerkt het van pdfTeX resulterende bestand en voegt de juiste effecten toe. Leuke resultaten kunnen op relatief eenvoudige wijze worden verkregen. Zie <http://www-sp.iti.informatik.tu-darmstadt.de/software/ppower4> voor enkele voorbeeldpresentaties met behulp van PPower4.

De effecten die PPower4 levert zijn vrij uitgebreid en houden bij creatief gebruik niet op bij het verschijnen van tekst. Dit kan samengaan met effecten bij het verschijnen: bijvoorbeeld in delen van letters. Ook kunnen bepaalde stukken tekst, bijvoorbeeld in een lijst één voor één worden uitgelicht. Zo kan het punt dat op dat moment wordt besproken ook visueel de aandacht krijgen.

Andere voordelen van het stap-voor-stap tonen zijn te vinden bij het geven van vergelijkingen. Door niet meteen meerdere stappen te geven, maar ze één voor één te tonen blijft de toehoorder geconcentreerd op de relevante stap en wordt niet afgeleid door de andere symbolen op de slide. Hetzelfde kan gerealiseerd worden met plaatjes, die in delen worden opgebouwd om bijvoorbeeld een algoritme te illustreren.

Een alternatief voor PPower4 is TeXPower. Omdat de maker van pdfscreen dit niet gebruikt levert het gebruiken van deze combinatie mogelijk af en toe problemen op.

Al met al is de combinatie pdfTeX/pdfscreen een redelijk toegankelijke manier om presentaties te maken die er goed uitzien en zo de inhoud visueel ondersteunen voor het publiek. Met PPower4 komt ook de liefhebber van effecten goed aan zijn trekken. Omdat pdfscreen meer kan dan alleen slides maken, is het soms net iets te uitgebreid en daardoor ingewikkelder in de omgang dan noodzakelijk was geweest.