

# Je proefschrift in LaTeX zetten

## Abstract

In dit artikel beschrijf ik hoe ik mijn proefschrift in LaTeX gezet heb. Ik ga in op mijn werkomgeving, de extra pakketten die ik gebruikte, de (lokale) truuks die ik in het bronbestand gebruikte en de problemen die ik ondervond (en oplossingen daarvoor).

## Keywords

Proefschrift, memoir, apacite, LaTeX

## Inleiding & werkomgeving

Een proefschrift schrijven is al behoorlijk uitdagende klus op zichzelf, al voordat je stil staat bij de manier waarop en het systeem waarin je je proefschrift wil gaan invoeren en laten zetten. De Sociale wetenschappen, waarin ik zelf recentelijk promoveerde, is een Microsoft gedomineerde markt waar het bepaald geen standaardkeuze is om je proefschrift in iets anders dan Microsoft Word te laten zetten. Al is het alleen maar om de samenwerking — via de inmiddels diep gewortelde ‘Review’-functie van Word<sup>1</sup> — niet te hinderen met bestandsformaten waarvan de gemiddelde Sociale wetenschapper geen kaas heeft gegeten.

Ook na de beslissing om alle “goede” advies in de wind te slaan is het bepaald geen makkie om je proefschrift in LaTeX te gaan uitvoeren. Het internet wemelt van vele goed bedoelde maar bijzonder fragmentarische webpagina’s vol adviezen die vaak op zeer lokale gebruiken en tradities georiënteerd zijn. Elke webpagina biedt wel een eigen style-file aan waarin alleen nog maar de inhoud gekopieerd hoeft te worden. Het converteren naar een bruikbaar formaat voor een Nederlandse universiteit is echter geen sinecure.

In dit artikel probeer ik enkele handgrepen te bieden voor de succesvolle promovendus die in het stadium van proefschrift schrijven is aangekomen en dit in LaTeX wil gaan doen. Dit is echter wel nadrukkelijk vanuit het perspectief van de Sociale wetenschappen gedaan. Het belangrijkste gevolg hiervan is dat andere normen gehanteerd worden bijvoorbeeld voor het zetten van de referenties in de tekst.

Daarnaast is het ook belangrijk te vermelden dat ik al jaren met LaTeX werk, maar dat ik eigenlijk nooit ingewikkelde klussen ter hand heb hoeven nemen. Dat wat onder LaTeX’s motorkap zit is voor mij niet te vergelijken met de binnenkant van mijn broekzak. Een

ervaren TeX-programmeur zal zich bij onderstaande ongetwijfeld wel eens op het hoofd willen krabben.

Voordat ik door het LaTeX-bronbestand van mijn proefschrift loop, geef ik eerst wat toelichting op de werkomgeving die ik hanteerde.

**Werkomgeving.** De keuze sec om een proefschrift in LaTeX te schrijven is er vaak eentje die al snel in de religieuze loopgraven uitgevochten wordt. Dit geldt net zo goed voor de gereedschappen waarmee je het werk uitvoert. Ik zal in het volgende vast wel een keer een gevoelige scheen raken; net zo vaak als, van de andere kant, een andere keuze achteraf meer gemak op had kunnen leveren.

Om te kunnen LaTeX’en heb je een distributie nodig waaruit je je classes en packages haalt. Voor mij is de TeXlive distributie al enige tijd het standaardstelsel op mijn computers. Dit hele artikel is dan ook gebaseerd op dit systeem; ik verwacht echter niet dat andere systemen zoals MikTeX volkomen anders werken. Om de meest belangrijke pakketten up to date te houden, installeerde ik de meest recente versie van de memoir class [1] en het apacite package [2] in mijn lokale TeX-boom. Als gebruiker van beamer voor mijn presentaties was ik al bekend met het PGF-package [4] waarmee je figuurtjes kan beschrijven in je LaTeX broncode. PGF is een variant op ps\_tricks speciaal geschikt voor het produceren van PDF-figuren.

Een goede editor is ook van groot belang waarbij je vooral op je persoonlijke ervaring en voorkeur moet letten. Zelf gebruik ik al jaren Vim en dit was dus de vanzelfsprekende keuze om mijn proefschrift in te typen. Extra plugins, zoals bijvoorbeeld de Vim-LaTeX suite [3] en een (engelstalige) spellingchecker zullen je werk wat aangenamer maken (zie bijvoorbeeld engspchk [5]<sup>2</sup>).

Tot slot installeerde ik ook nog Cygwin op mijn Windows XP machine. In eerste instantie omdat een Bash-shell toch wat efficiënter werkt dan de standaard XP command prompt; een automatische toetsencombi om mijn proefschrift in Vim te laten te compileren heb ik nooit echt goed voor elkaar gekregen. Bovendien bleek, in tweede instantie, dat het compileren van een proefschrift flink wat tussenstappen vereiste omdat ik natuurlijk een referentielijst moest opnemen (al dan niet geordend per hoofdstuk) en een auteursindex ook erg gewenst bleek. Het automatiseren van het compi-

leren leek mij veel gemakkelijker in een Linux-achtige omgeving dan onder Windows.

Om de compilatieproblemen op te lossen bestaan er vele verschillende oplossingen die met behulp van Google gemakkelijk gevonden kunnen worden. De oplossingen variëren van relatief simpele batchfiles tot zeer geavanceerde Ruby of Perl scripts en Makefiles. De Makefile die ik koos behoefde ook nog enige aanpassing, omdat ik via het `apacite` package een auteurs-index wilde genereren die, zo bleek, in extra `.and` en `.adx` files resulteerde waarmee de oorspronkelijke Makefile geen raad wist. Uiteindelijk kon ik met één commando ofwel een DVI-file ofwel een PDF-file laten compileren. De gewijzigde Makefile is terug te vinden op mijn website [8]; dat geldt ook voor het basisbronbestand van mijn proefschrift en andere files die te lang zijn om in dit artikel integraal op te nemen.

### Walkthrough

Een belangrijk aspect van mijn proefschrift is het feit dat ik de memoir class gebruik heb als machine tussen mijn tekst en  $\text{LaTeX}$ . Eigenlijk wist ik niet van het bestaan van deze handige klasse af — Piet van Oostrum's hint over deze klasse in een eerder Maps artikel viel me pas achteraf op — totdat ik wat aan het rondsnuffelen was over het opmaken van boeken en daarbij het manual van deze klasse als een van de hits opdoek. Oorspronkelijk had ik het proefschrift simpelweg willen maken met behulp van de standaard book klasse, maar het aanpassen van de (standaard) opmaak leek me een behoorlijke klus, zeker omdat ik die internals van  $\text{T}_{\text{E}}\text{X}$  en  $\text{LaTeX}$  nog onder de knie moest krijgen. Memoir biedt echter de ideale oplossing als een extra niveau tussen  $\text{T}_{\text{E}}\text{X}$ ,  $\text{LaTeX}$  en de brontekst. Zo ongeveer elk oorspronkelijk commando van  $\text{LaTeX}$  wordt gehedefinieerd en voorzien van allerlei hooks die veel gemakkelijker zijn aan te passen in je bronbestand dan dat het is om de oorspronkelijke  $\text{LaTeX}$ -commando's aan te passen. Iemand die zich herkent in mijn plaats in de  $\text{LaTeX}$ -wereld kan ik deze klasse zeker aanraden!

```
\documentclass[a4paper,twoside,12pt,
openright,final]{memoir}
```

Het eerste commando<sup>3</sup> van een memoir document ziet er niet veel anders uit dan dat van een regulier book- of article-klasse document. De optie `final` heeft bij memoir, net als in alle standaard  $\text{LaTeX}$  klassen, de `partneroptie` `draft`. Wanneer je `draft` activeert, dan worden alle overfull hboxes gemarkeerd met een duidelijk zwart balkje in de marge en wordt het invoegen van figuren overgeslagen (althans, ze worden in de PDF/DVI alleen met een kader gemarkeerd). Hierdoor wordt het compileerproces soms merkbaar sneller afgerond.

**Standaard packages.** Na het inladen van de class de standaardpakketten. De echt standaard zaken heb ik niet genoemd (`babel`, `graphicx`, etc.); let echter op dat sommige pakketten in een bepaalde volgorde geladen moeten worden!

```
\usepackage{pgf}
\usepackage{booktabs}
\usepackage{europs}
\usepackage{wrapfig}
```

`pgf` laadt de nodige definities in voor de voorbereiding van het maken van PDF-tekeningen in je  $\text{LaTeX}$ -bronbestand. `booktabs` gebruikte ik om de tabellen in het proefschrift wat eleganter te maken. De voorname wijziging die dit pakket doorvoert betreft de spatiering tussen lijnen en tekst. Daarnaast heeft het ingebouwde mogelijkheden om dikkere en dunnere lijnen te gebruiken.

Met `europs` wilde ik voorkomen dat mijn proefschrift vol zou komen te staan met “EUR” in plaats van €. Het `wrapfig` package heb ik op het allerlaatste moment opgenomen om één figuur die uiteindelijk kleiner uitviel dan de overige op een elegante positie in de lopende tekst in te laten vloeien. Er bestaan meerdere pakketten met deze doelstelling, maar `wrapfig` (b)leek op dat moment de simpelste keuze die aan mijn verwachtingen voldeed. Alle andere figuren heb ik steeds (uitsluitend) boven aan de pagina geplaatst (of op groot formaat op een aparte pagina) en de schaling zodanig aangepast dat er een prettige evenwicht ontstond tussen de tekst en de figuur met caption.

**Packages met aanpassingen.** Na deze standaardpakketten die weinig aanpassing nodig hadden heb ik de wat complexere pakketten met persoonlijke modificaties opgenomen.

Het is in mijn instituut gebruikelijk om een proefschrift te baseren op een serie van (gepubliceerde) artikelen. Om de relatie tussen artikel en hoofdstuk te definiëren wordt de hoofdstuktitel van een voetnoot voorzien die aangeeft uit welk(e) artikel(en) de tekst komt. Ik wilde hiervoor een aparte voetnotemark gebruiken. Gewoon footnotes gebruiken en daarna doornummeren (eerste echte voetnoot wordt dan nr. 2) vond ik lelijk. Het nummer resetten was ook geen optie (twee noten met nr. 1?).

Ik koos voor een dagger, maar dan wel via standaard commando's omdat ik automatisch een uniforme voetnootopmaak wilde doorvoeren. Voetnoten zijn echter fragile waardoor in de sectioning commando's een `\protect` toegevoegd moet worden. Een extra complicatie was dat ik een zeer strakke cq. minimalistische (recto) openingspagina voor mijn hoofdstukken wilde creëren. De voetnoten moesten op de volgende verso-pagina belanden. Twee websites [6, 7] en het

footmisc package boden uitkomst.

```
\usepackage[stable]{footmisc}
\newcommand{\setssymbolfootnote}{%
  \def\thefootnote{\fnsymbol{footnote}}
\long\def\symbolsymbolfootnotemark[#1]{\begingroup%
\protect\setssymbolfootnote\footnotemark[#1]%
\endgroup}
\long\def\symbolsymbolfootnotetext[#1]#2{\begingroup%
\protect\setssymbolfootnote\footnotetext[#1]%
{#2}\endgroup}
```

De nieuwe commando's komen van bovengenoemde websites waar ze eigenlijk gedefinieerd worden als (uitsluitend) een reguliere `\footnote`. Om de footnotemark en footnotetext te scheiden creëerde ik dezelfde varianten, maar dan met een symbol als footnotemark.

Het laatste en belangrijkste package dat ik moest aanpassen was `apacite` van Erik Meijers. Citaties volgens de voorschriften van de American Psychological Association (APA) is de standaard in de Sociale wetenschappen. In eerste instantie gebruikte ik nog een behoorlijk oude versie, maar afgelopen jaar bracht Erik een tussenversie uit met behoorlijke uitbreidingen in functionaliteit.

```
\usepackage[emindex,bibnewpage,hyper]{apacite}
\renewcommand{\bibliographyprenote}{%
  {\vspace{1em}}
\renewcommand{\bibliographytypesize}{%
  {\footnotesize}}
\newcommand{\Dutchvon}[2]{#2}
\renewcommand{\B0thers}[1]{et al.}}
```

Volgens het manual moet dit pakket liefst als allerlaatste package geladen worden, maar in elk geval pas achter Babel. De opties geven aan dat ik een auteursindex wil die in de TOC opgenomen wordt; vergeleken met de andere index-opties zoals `tocindex` verzorgt `emindex` ook nog eens de titel van de auteursindex. Zet deze index op een nieuwe pagina met behulp van `bibnewpage` en 'misbruik' het `bibliographyprenote` om wat meer ruimte tussen header en tekst te krijgen. Gebruik de optie `hyper` voor compatibiliteit met het `hyperref` pakket dat ik wil gebruiken voor de online versie van mijn proefschrift.<sup>4</sup>

`bibliographytypesize` is een herdefinitie om de referentielijst in een kleiner font te zetten. Voor de auteursindex biedt `apacite` deze haak helaas niet waardoor ik de style file moest aanpassen door in de definitie van de `emindex` optie een `\footnotesize` in te voegen.

Het `Dutchvon` commando is een hack van Erik zelf waarmee je de referentielijst op een wat Nederlandse manier kan sorteren; deze sortering is dan echter niet meer strikt volgens de APA norm. Dit commando

vereist echter ook de nodige aanpassingen in de `.bib`-files en nog een verandering van `Dutchvon` net voor de referentielijst. Zie de documentatie van `apacite` voor meer info.

De laatste herdefiniëring (`\B0thers`), tot slot van `apacite`, betreft een bug die ik ontdekte bij zeer fijnkorrelige inspecties van LaTeX's output. Volgens de normen van de American Psychological Association (APA) mag je bij een artikel met meer dan drie auteurs vanaf de tweede citatie in de tekst de 2e auteurs (en verder) vervangen door "et al.". Het pakket van Erik regelt dat helemaal automagisch, maar het bleek dat de punt achter 'et al' door LaTeX geïnterpreteerd werd als een sentence-ending dot met extra spatiering. Dit probleem is met deze herdefinitie op te lossen, maar heeft als nadeel dat je je zinnen niet meer kan eindigen met een citatie, omdat hier dan ook de sentence-ending spacing zal verdwijnen. Ik heb me aan deze beperking kunnen houden, maar ik heb van Erik ook een stukje code waarin zelfs dit probleem met een extra commando opgelost wordt. Deze code heb ik op mijn website geplaatst [8]. Het resultaat van dit alles is opgenomen in de figuur met de openingspagina van mijn referentielijst.

Als laatste pakket heb ik `hyperref` ingevoegd om hyperlink navigatie toe te voegen voor de online versie van het proefschrift [8].

```
\usepackage[pdftex,pagebackref=true,
  linktocpage=true,breaklinks=true,
  bookmarks=true]{hyperref}
\hypersetup{%
  pdftitle={Congruency versus strategic
  effects in multimodal affective
  picture categorization},
  pdfauthor={Paul M. C. Lemmens},
  pdfsubject={Cognitive psychology and
  human-factors research},
  pdfcreator={PDFLaTeX (TeXLive 2003)
  and Vim},
  pdfproducer={PrintPartners Ipskamp,
  Enschede, The Netherlands},
  pdfkeywords={Stimulus-response
  compatibility, affective
  compatibility, affective-computing
  research},
  pdfpagemode={UseThumbs},
  pdfstartview=FitV,
  pdfpagelayout=SinglePage
}
\usepackage{memhfixc}
```

Ik heb hier niet lang over nagedacht en een setup overgenomen van een website (of misschien zelfs de officiële documentatie; ik weet het niet meer) die voor mij werkte en acceptabele resultaten gaf. Met de op-

ties bij het aanroepen van de style file geef ik aan dat LaTeX eventuele links die over een regelafbreep punt heen gaan ook daadwerkelijk mag afbreken, dat er bookmarks aangemaakt moeten worden en dat er vanaf de TOC gelinkt wordt naar de openingspagina's van de hoofdstukken.

De meeste parameters van het `hypersetup` commando zullen ook wel voor zich spreken. De laatste drie geven aan hoe de PDF-lezer het PDF-bestand moet openen en weergeven. In dit geval met de thumbnails aan de linker kant, de pagina wordt verticaal passend op het scherm weergeven met één pagina per scherm. Jammer genoeg is de '2-page facing' optie van Acrobat geen optie, want daarvoor kon ik geen instellingen vinden.<sup>5</sup>

Het extra package `memhfixc` is een hack van de maker van `memoir` om dit laatste pakket compatibel te maken met `hyperref`. `Memoir` verandert zaken aan bijvoorbeeld de `\chapternumberline` commando's die `hyperref` niet kan hanteren. Dit extra package lost die problemen op.

Overigens heb ik de hyperlinks pas toegevoegd toen het officiële manuscript al lang bij de drukker lag. Dit simpelweg om 100% zeker te zijn dat de kaders die `hyperref` om de links tekent niet in de gedrukte versie van het proefschrift terecht zouden kunnen komen.

**Persoonlijke macro's.** Door `memoir` te gebruiken had ik uiteindelijk vrijwel geen complexe TeX-code meer nodig. Mijn macro's bestonden vooral uit een aantal simpele commando's die het typen van tekst wat gemakkelijker maken.

```
\newcommand{\schuinig}[1]{\textsl{#1}}
\newcommand{\Stim}{\schuinig{S}}
\newcommand{\Sr}{\Stim\ensuremath{r}}
\newcommand{\Si}{\Stim\ensuremath{i}}
\newcommand{\Resp}{\schuinig{R}}
\newcommand{\Response}{\Resp}
\newcommand{\R}{\Resp}
\newcommand{\SR}{\Stim--\Resp}
\newcommand{\SrR}{\Sr\,--\,R}
\newcommand{\SiR}{\Si\,--\,R}
\newcommand{\SiSr}{\Si\,--\,Sr}
```

Ik nam een omgeving over uit de `apa.cls` om opsommingen in de lopende tekst mogelijk te maken.

```
\newcounter{lappieAPAenum}
\def\seriate{\begingroup%
  \setcounter{lappieAPAenum}{0}%
  \def\item{\addtocounter{lappieAPAenum}{1}%
    (\alph{lappieAPAenum})\space}%
  \ignorespaces}
\def\endseriate{\endgroup}
```

Mijn persoonlijke trots is nog altijd het `\marginnote` commando dat ik zelf in elkaar gestoken heb met

veel proberen en optimaliseren. Omdat mijn werkmodus vaak onderbroken wordt door gedachtes die ik heb over stukken tekst die ik typ — wijzigingen, alternatieve formuleringen, aandachtspunten, et cetera — gebruikte ik vaak het `marginpar` commando om die gedachtes snel vast te leggen. Al snel werden dit opmerkingen die ik zelf al na twee dagen niet meer snapte omdat de context ontbrak.

```
\newcounter{marginnotecounter}%
\setcounter{marginnotecounter}{1}%
\newcommand{\marginnote}[1]{%
\def\baselinestretch{0.9}%
\textbf{(\themarginnotecounter)}%
\marginpar{%
\parbox[t]{20mm}%
{\raggedright\tiny\textsf{%
(\themarginnotecounter) #1}}}%
\addtocounter{marginnotecounter}{1}}%
```

Dus definieerde ik een nieuwe teller die een positie in de tekst markeerde en vervolgens in de marge de noot en teller plaatste. Maar wanneer ik, wederom als verplichting van de APA, m'n regelafstand verdubbelde voor manuscripten, dan werd die regelafstand in de marginnoten meegenomen. In principe van weinig belang, omdat die opmerkingen toch vanzelf zouden moeten verdwijnen voordat een manuscript definitief zou mogen worden, maar het leverde visueel erg hinderlijke DVI's op. Daarom voegde ik in de laatste versie een lokale `baselinestretch` aanpassing door en verkleinde ik het font. Een voorbeeld van een marginoot is te zien in figuur 1, aan het eind van dit verhaal.

**Opmaak van het manuscript.** Toen de functionaliteit van het bronbestand eindelijk geregeld was, kon ik aan de opmaak beginnen. Eigenlijk had ik dit alles natuurlijk in een stylefile moeten gooien, maar dat is een stap waar ik nooit meer aan toe gekomen ben. Veel van deze aanpassingen heb ik pas doorgevoerd toen het manuscript inhoudelijk helemaal af was en tot drie keer toe gecontroleerd was op typefouten en andere onhebbelijkheden. Pas dan heb je je manuscript namelijk helemaal af zodat je echt de fijne details van de opmaak kan gaan regelen. Maak niet de fout om deze fijnkorrelige details te vroeg te willen regelen, want dan kom je uiteindelijk in een vicieuze cirkel terecht — ik spreek uit ervaring!

```
\addtolength{\textheight}{17pt}
\checkandfixthelayout
```

De eerste aanpassing die ik doorvoerde was een pragmatische: ik kwam op 145 pagina's uit wat één pagina meer was dan dichtsbijzijnde 16-voud<sup>6</sup>. Het verhogen van de tekstheight met 17 punten was precies genoeg om één pagina van het totaal aantal af te snoe-

pen. Het commando `\checkandfixthelayout` is een min of meer verplicht memoir commando dat na zulke wijzigingen er voor zorgt dat allerlei gerelateerde parameters op de juiste manier aangepast worden. Als ik me goed herinner wordt er ongeveer een heel hoofdstuk in het memoir manual aan de paginaopmaak gewijd; je zult het mij niet kwalijk nemen dat ik dat hier niet ga herhalen!

```
\usepackage{setspace}
\newcommand{\BaseLineStretch}{1.10}
\setstretch{\BaseLineStretch}
```

Binnen de Sociale wetenschappen, maar vermoedelijk ook binnen andere richtingen is het gebruikelijk om manuscripten met een dubbele regelafstand in te dienen. Sommige promovendi hanteren deze norm zelfs bij het opmaken van de uiteindelijk (te drukken) versie van hun proefschrift, maar dat vond ik niet mooi. Van de andere kant vond ik de leesbaarheid met regelafstand van 1 te laag en na lang proberen en laten vergelijken door “naïve” lezers kwam ik op een afstand van 1.10 of 1.15 uit. Ik hield de 1.15 achter de hand voor het geval dat ik het pagina-aantal zou moeten verhogen naar een 16-voud.

Door trial en error ontdekte ik overigens een eigenaardige, maar achteraf gezien logische, interactie tussen het veranderen van de baselinestretch en de `\[-3mm]` commando's die ik gebruikte in een PGF-plaatje (zie figuur 2.2 in mijn proefschrift.) Voor een artikelversie van een bepaald hoofdstuk, dat op dubbele regelafstand moest, had ik de negatieve afstand in het `\[` commando nodig om de tekst in het PGF-plaatje (verticaal) juist uitgelijnd te krijgen. Toen ik de regelafstand weer verkleinde voor het proefschrift was er natuurlijk opeens teveel negatieve afstand. Het duurde even voordat ik deze interactie in de gaten had.

```
\usepackage{palatino}
\usepackage{pxfonts}
```

Iedereen die vaker LaTeX heeft gezien of gebruikt herkent het in één oogopslag: het is ofwel de pagina-indeling ofwel het font dat een LaTeX-gebruiker verdraagt. De pagina-indeling was al voldoende anders dus bleef alleen het font over om nog te wijzigen. Ik hanteerde twee mogelijke varianten: Bookman of Palatino. Veel van mijn proeflezers vonden de sample van pagina's in Bookman leuker en prettiger lezen. Bookman ziet er iets dikker uit wat wel een bepaalde aantrekking had. Zelf was ik echter nog net iets meer gecharmeerd van Palatino. Dat feit gecombineerd met een forse verhoging van het totaal aantal pagina's bij gebruik van Bookman deed de keuze uiteindelijk op Palatino vallen.

Gelukkig merkte ik nog net op tijd dat het standaard package palatino de fonts in mathmode niet mee

deed veranderen. Via TeX-NL werden de suggesties mathpazo en pxfonts gegeven. Puur op visuele basis koos ik uiteindelijk voor pxfonts om het Palatino font mooi in de mathmode door te laten lopen.<sup>7</sup> Ik heb geen gekke sprongen meer gedaan om nog de oldstyle cijfers te realiseren (al had ik dat misschien wel mooi gevonden), want daarvoor was de tijdsdruk te groot.

Het volgende aandachtspunt voor de opmaak was de openingspagina van de hoofdstukken. Zoals ik al eerder aangaf wilde ik deze zo clean mogelijk houden: een cijfer, de titel en eventueel nog een quote in een benedenhoekje.

```
\makeatletter
\makechapterstyle{companion}{%
  \renewcommand{\afterchaptertitle}{%
    \thispagestyle{empty}}
  \renewcommand{\chapnamefont}{%
    \normalfont\LARGE\scshape}
  \renewcommand{\printchaptername}{%
    \raggedleft\chapnamefont \@chapapp}
  \renewcommand{\chapnumfont}{\normalfont\Huge}
  \setlength{\chapindent}{\marginparsep}
  \addtolength{\chapindent}{\marginparwidth}
  \renewcommand{\printchaptertitle}[1]{%
    \begin{adjustwidth}{-{\chapindent}}
      \raggedleft \chaptitelfont ##1\par\nobreak
    \end{adjustwidth}}
}%
\makeatother
\pagestyle{companion}
\chapterstyle{companion}
```

Ik had al besloten dat ik de door memoir voorgedefinieerde companion-emulatie voor pagestyle en chapterstyle wilde gebruiken. Het paginanummer op de openingspagina van het hoofdstuk wilde ik echter verwijderen. Het lukte me niet om dit met een of andere hook van memoir te corrigeren, dus besloot ik simpelweg de gehele definitie uit de classfile over te nemen en naar believen aan te passen. Let op de `\makeatletter` en `\makeatother` die nodig zijn wanneer je een `\@`-commando gebruikt. Als ik een stylefile gebruikt had waarin deze set van commando's was opgenomen, dan had ik de `\makeat*` commando's achterwege kunnen laten. Daarna kon ik veilig de pagestyle en chapterstyle voor de rest van het proefschrift vastleggen. Figuur 2 laat een voorbeeld zien van een hoofdstuk-pagina.

Toen ik besloot in de TOC uitsluitend de hoofdstukken aan te geven (`\maxtocdepth{chapter}`), bleek dit erg lelijk te worden. Het vet van het nummer en de titel van de hoofdstukken werd niet onderbroken door lijstjes van niet-vet gedrukte sections in die hoofdstukken. Met behulp van `\cftchapterpresnum` en `\cftchapterfont` kon ik het nummer van het hoofd-

stuk wel in het vet krijgen, maar bleef de titel van het hoofdstuk netjes in normale tekst staan.

```
\renewcommand{\cftchapterpresnum}{\bfseries}
\renewcommand{\cftchapterfont}{-}
\maxtocdepth{chapter}
```

Natuurlijk is een goed proefschrift doorspekt van min of meer diepzinnige quotes van min of meer beroemde mensen. Memoir biedt daarvoor het epigraph mechanisme. Voor mijn proefschrift paste ik de default breedte en positie aan.

```
\setlength{\epigraphwidth}{0.5\textwidth}
\epigraphposition{flushleft}
```

Voor het zetten van een aantal (sets van) plaatjes gebruikte ik in eerste instantie het subfig pakket. Toen ik echter met de opmaak van de captions ging spelen bleek dat subfig en memoir elkaar niet aardig vinden. Aangezien memoir en de wijzigingen aan de caption prioriteit hadden moest subfig wijken. Het manual van subfig (of een ander subfig-achtige) gaf een goede hint om een bepaalde opmaak met subplaatjes ook met gewone TeX-commando's voor elkaar te krijgen. Daarna was het nog slechts een kwestie van de juiste hooks aanpassen om het font te verkleinen en (dus ook) de afstand tussen de broodtekst en de captions te verkleinen.

```
\captionnamefont{\footnotesize}
\captiontitlefont{\footnotesize}
\setlength{\textfloatsep}{35pt}
```

**Inhoud.** Met deze aanpassingen kon ik het opmaken van het manuscript afsluiten. In het vervolg van dit artikel wil ik een aantal (memoir) commando's uitlechten die ik in de inhoudelijke tekst van het proefschrift kon gebruiken. Op de standaardzaken zoals `\title` `\author` etc. zal ik niet ingaan; net zomin op de `\frontmatter`-commando's. Het verdient wel de aanbeveling om de theoretische hoofdstukken van het memoir manual eens door te lezen over de indeling en de opzet van de zogenaamde half-title page, frontispiece/copyright page enz.

Toen ik, op een verder lege pagina, de copyright mededelingen volledig aan de onderkant van de pagina wilde plaatsen, ondervond ik dat `\vfill` niet zomaar de grootst mogelijke verticale afstand opeet. Voor mij was het een verrassing dat er 'tekst' voor dit commando moet staan. De oplossing dient zich dan relatief eenvoudig aan: `\ \vfill`".

Ik merkte ook door trial and error dat wanneer ik de lange titel van het proefschrift in een groter font wilde zetten op de pagina, dat dan (klaarblijkelijk) de regelafstand niet aangepast wordt. Het

eindresultaat is dan dus een aantal regels dat (verticaal) veel te kort op elkaar staat. Hier kwam het `setspace` package goed van pas. Een beetje tweakten leverde een regelafstand van 1.35 op voor een `\textbf{\Large{titeltekst}}`.

Bij het bekijken van de titelcode kwam ik nog een zogenaamde FIXME tegen. Het is een markering die ik gebruik om incomplete stukken tekst, broncode of anderszins te markeren voor latere analyse.

```
%%FIXME: uitlijning op 1 regel!
een wetenschappelijke proeve op het gebied %%
van de Sociale Wetenschappen
```

In dit geval had ik al geconstateerd dat in de PDF het woord Wetenschappen een regel lager stond dan de rest van de zin. Klaarblijkelijk heb ik deze FIXME uiteindelijk gemist bij mijn controles (zie de figuur met het titelblad van mijn proefschrift[8]) en natuurlijk pas net na het accorderen van de drukproeven weer opgemerkt. Om dit soort situaties te voorkomen heb ik eens een pakket gevonden dat een commando implementeerde dat vergelijkbaar is met mijn `\marginnote`, met de aanvulling dat aanwezigheid van marginnotes het compileerproces onderbreekt. De naam en plaats van dit pakket heb ik helaas niet meer kunnen achterhalen.

Na deze (door de universiteit verplichte) pagina's dan toch eindelijk de wetenschappelijk inhoud. Elk hoofdstuk heb ik met een `\include` in het hoofdbestand opgenomen. Hieronder licht ik het voor dit publiek interessantste gedeelte uit.

```
\chapter[Stimulus--Response Compatibility and
Affective Computing: A Review][Stimulus--%
Response Compatibility and Affective
Computing]{Stimulus--Response Compatibility
and Affective Computing: A Review}
\symbolfootnotemark[2]\label{ch:backgr}%
\label{ch:background}
```

Een belangrijke reden om memoir te gebruiken, behalve de handige hooks, was het `\chapter`-commando dat maar liefst drie argumenten kent. De standaard LaTeX-versie kent er maar twee: een verplicht argument en een optioneel argument voor in TOC. Het verplichte argument wordt echter ook gebruikt in de header van de pagina. Meestal dus veel te lang, waardoor de tekst gaat overlappen. Hierboven dus drie titels: de eerste voor in de TOC, de tweede voor in de header en de derde als echte titel van het hoofdstuk. In dit derde argument kom je ook weer het `\symbolfootnotemark`-commando tegen dat ik eerder definieerde om een lege, minimalistische hoofdstukopenerpagina te creëren.

```
%% Epigraph!?
\vfill
```

```
\epigraph{I often say ...}%
  {\textit{Lord Kelvin (1824--1907)}}
```

```
\cleartoverso
\section*{Abstract}
```

```
\input{intro-review-abstract}
```

```
\symbolfootnotetext[2]{This chapter is
  submitted as: ...}
```

```
\cleartorecto
\section{Introduction}
```

```
\input{background}
```

Met het `\cleartoverso` gaan we naar de ‘achterkant’ van deze openingspagina (althans in de uiteindelijke dubbelzijdige versie van het proefschrift) waar we het abstract afdrukken, de voetnoot plaatsen en daarna weer doorschuiven naar de volgende rechterpagina om het hoofdstuk (inhoudelijk) te beginnen. Het resultaat van deze LaTeX-code (incl. het `\chapter[] [] {}`-commando in de vorige paragraaf) is te zien in de figuur met daarin de openingspagina van hoofdstuk 2 van mijn proefschrift.

In een bepaald hoofdstuk wilde ik naderhand nog wat controle-studies toevoegen die echt specifiek bij dit hoofdstuk hoorden. Ik verwachtte dat ik een rare constructie zou moeten gaan bedenken om de appendices niet meer numeriek te laten nummeren, maar ook hier kwam memoir weer als onverwacht behulpzaam om de hoek.

```
\begin{subappendices}
\end{subappendices}
```

De subappendices omgeving kun je gebruiken om (eenmalig) één of meerdere appendices achter een hoofdstuk in te voegen, in plaats van op het einde van een boek. Het zorgt voor een andere naamgeving (“Appendix”) en een alfabetisch telling (bijv. “4B”) van de secties. Met behulp van het `\section`-commando kun je verschillende appendices opnemen in de subappendices omgeving.

### Backmatter

En dan na heel veel typewerk en het gebruikelijke schrappen kom je aan de laatste loodjes. De backmatter van je proefschrift. Omdat dit een apart gedeelte is van het document heb ik hier veiligheidshalve de `\pagestyle{companion}` en `\chapterstyle{companion}` herhaald.

Omdat ik wilde dat de referentielijst (via `\renewcommand{\refname}{References}`) wat kleiner gezet werd en wat gecompriëerder op de pagina moest

komen (denk weer aan die veelvoud van 16 pagina’s) heb ik enkele van de nieuwe apacite-hooks gebruikt.

```
\setstretch{1.0}
\setlength{\bibindent}{-1em}
\setlength{\bibitemsep}{+\parsep}
```

Als toetje ook nog een auteursindex omdat ik geen referentielijsten per hoofdstuk heb geprint: `\printindex[autx]`. Om voor deze lijst dezelfde regelafstand en met name fontgrootte te kunnen handhaven moest ik helaas wél in de `.sty`-file van apacite hacken.

### Lastige figuur

Ik moest een tamelijk complexe figuur maken om experimentele resultaten te verhelderen; het complexe betrof vooral het inlezen en plaatsen van mijn stimuli in die figuur. Mijn statistiek deed ik voornamelijk in R [9] dat alleen `pixmap` kent als een grafisch formaat dat binnen een gegenereerde figuur geplaatst kan worden.<sup>8</sup> Wegschrijven van een figuur in verschillende formaten is in R geen probleem: PNG, JPG, EPS en PDF behoren alle tot de mogelijkheden. De inhoud van de figuur bestond uit geïmporteerde plaatjes, tekst en gegenereerd grafisch werk (zie pagina 70 van mijn proefschrift [8]).

Nadeel van wegschrijven in PNG is dat de figuur niet echt lekker meer wil op-/neerschalen met name omdat de fonts dan lelijk worden (idem voor JPG). PDF was op dat moment geen optie omdat het me niet lukte om de fonts in de resulterende file op te nemen; dat betekende dat de PDF van mijn hele proefschrift niet door de preflight van Adobe zou komen. EPS bleek echter ook niet handig omdat al die kleine plaatjes die ik invoegde uiteindelijk tot een EPS bestand van ruim 50Mb samengevoegd werden. Dat werkte niet handig met het verder verwerken van de PDF van het hele proefschrift.

Via TEX-NL kreeg ik een groot aantal tips toegezonden die varieerden van compressietechnieken tot allerlei conversiepaden. Gelukkig bleek ik deze oplossingen niet meer nodig te hebben omdat het probleem zich onverwacht makkelijk liet versimpelen. Wanneer ik mijn plaatje via de menu-interface van R wegschreef werd het een factor 10 kleiner dan wanneer ik het volledig via de programmeertaal liet genereren; Het waarom hierachter ontgaat me (nog steeds). De conversie van EPS naar PDF was toen natuurlijk zo geregeld en had ik een werkbaar plaatje mét fonts in mijn proefschrift staan.

### Conclusies

Uiteindelijk is het maken van mijn proefschrift een behoorlijk heftig leerproces geweest en dan niet alleen op TeXnisch en wetenschappelijk gebied, maar ook op

andere aspecten. Het TeX-en was op momenten wel het leukste om mee bezig te zijn, al heb ik er uiteindelijk toch weer minder tijd aan kunnen besteden dan ik me aan het begin van het project had voorgenomen.

Diegene die het eindresultaat nog niet voor zijn geestesoog kan voorstellen wil ik graag doorverwijzen naar mijn persoonlijke website [8].

### Referenties

- [ 1 ] Wilson, P. (2005). *The Memoir Class for Configurable Typesetting*. Available from CTAN via [macros/latex/contrib/memoir/](http://macros/latex/contrib/memoir/).
- [ 2 ] Meijer, E. (2005). *The apacite package*. Available from CTAN via [macros/latex/contrib/apacite/](http://macros/latex/contrib/apacite/).
- [ 3 ] Avadhanula, S., Machowski, M. & Fisher, B. (2005). *The Vim-LaTeX suite*. Available via <http://vim-latex.sourceforge.net/>.
- [ 4 ] Tantau, T. (2005). *The PGF package*. Available via [https://sourceforge.net/project/showfiles.php?group\\_id=92412](https://sourceforge.net/project/showfiles.php?group_id=92412).
- [ 5 ] Campbell, C. (2005). *Engspchk Spelling Checker*. Available via [http://www.vim.org/scripts/script.php?script\\_id=195](http://www.vim.org/scripts/script.php?script_id=195).
- [ 6 ] <http://help-csli.stanford.edu/tex/latex-footnotes.shtml>.
- [ 7 ] <http://listserv.surfnet.nl/scripts/wa.exe?A2=ind0501&L=tex-nl&F=&S=&P=2474>.
- [ 8 ] <http://www.paul-lemmens.nl/maps>.
- [ 9 ] R Development Core Team (2006). *R: A language and environment for statistical computing*. <http://www.r-project.org/>.

### Voetnoten

1. Sinds versie 6.0 heeft Acrobat, de volledige suite, de mogelijkheid om PDF-bestanden van commentaar te voorzien en tot op zekere hoogte te bewerken. In hoeverre de niet-commerciële producten deze mogelijkheid ook (gaan) bieden is mij onbekend.
2. De recentste versie van Vim, versie 7.0, heeft een ingebouwde spellingchecker waarvoor nederlandse woordenbestanden eenvoudig toegevoegd kunnen worden.
3. Om de opmaak van dit artikel enigszins leesbaar te houden heb ik veel LaTeX-commando's handmatig afgebroken zodat ze in de kolombreedte passen. Regeleindes gemarkeerd met twee procenttekens behoren op één regel.
4. Hyperref heb ik overigens pas na het goedkeuren en drukken van het proefschrift toegevoegd, om te voorkomen dat de boxen en toevoegingen die dit pakket realiseert (toch) onverhoopt in de gedrukte versie zouden belanden. In hoeverre deze angst gegrond is weet ik echter niet.
5. De Mapsredactie wees me op de optie `pdfpagelayout=TwoColumnRight`, die de 2-page facing optie van Acrobat kan aanzetten.
6. Alle drukkers werken met grote vellen waarop 8 pagina's op de voorkant en 8 pagina's op de achterkant gedrukt worden. Deze vellen worden vervolgens automatisch gevouwen en gesneden tot "die kleine miniboekjes" (katernen) die je in elk gedrukt boekwerk tegenkomt.
7. De redactie van de Maps wees mij op het feit dat pxfonts een volledige vervanger is van palatino, waarbij (ook nog) de cijfers in mathmode door een palatino variant vervangen worden.
8. Vermoedelijk zijn er ook andere cq. betere oplossingen te vinden, maar daarvoor had ik op dat moment geen tijd over.

Paul Lemmens

paul (at) paul-lemmens (dot) nl



## 2.1 Introduction

Traditionally cognitive science and experimental psychology have investigated information processing within individuals (e.g. see Donders, 1868/1969; Kornblum et al., 1990; Sternberg, 1969). Since recently, however, the research investigating interindividual information-processing is gaining momentum (e.g. see Frith & Wolpert, 2003; Van Schie, Mars, Coles, & Bekkering, 2004)(1). Interindividual communication and information processing has been one of the pillars of human-factors research to improve human-computer interaction by trying to make it more like human-human interaction, for instance by using natural-language dialogue. Although often ignored, an essential component of human-human interaction is emotion: A wealth of non-verbal information can be conveyed by gestures or prosody. Even without verbal communication, humans often feel an immediate like or dislike for a person they meet for the first time, or, even between life-long partners communication can ignite a fierce debate with only a few sentences carrying an unexpected and (usually) unintended emotional charge. The success of emoticons in human-computer-human communication (email, chatting and online-presence software) corroborates the important role of emotion in human-human and human-computer interaction.

(1) Check if I have somewhat less neuroscientific literature on this topic.

Originally human-computer interaction (HCI) did not involve emotion at all; the computer was seen as a binary machine devoid of any possibilities to position its responses between the extremes of zero and one and users simply had to adapt to that situation (Brave & Nass, 2003). A similar perspective was taken in experimental psychology where emotion (hot cognition) and reason (cold cognition) were long taken to be completely separated (e.g., Sorrentino & Higgins, 1986). Picard (1997), therefore, argued that perception and appropriate presentation of affect by a computer might facilitate interaction with users and improve user performance in, as well as user satisfaction of, such an interface. She proposed to investigate Affective Computing which she defined as 'all computing that relates to, arises from, or deliberately influences emotions'.

Perhaps one of the reasons why hot and cold cognition have been viewed as separate is that the former is usually associated with responses on the level of the autonomous nervous system (e.g. heart rate, blood-volume pressure, skin conductivity, Brave & Nass, 2003) whereas the latter is associated with higher cognitive mechanisms. Most experiments

**Figuur 1.** Voorbeeld van een margenoot

## CHAPTER 2

**Stimulus–Response Compatibility and  
Affective Computing: A Review<sup>†</sup>**

I often say that when you can measure  
what you are speaking about, and  
express it in numbers, you know  
something about it; but when you  
cannot measure it, when you cannot  
express it in numbers, your knowledge  
is of a meagre and unsatisfactory kind.

---

*Lord Kelvin (1824–1907)*

**Figuur 2.** Voorbeeld van een hoofdstuk-pagina