



FACULTEIT INDUSTRIËLE WETENSCHAPPEN
AFSTUDEERRICHTING: INFORMATICA

Academiejaar 2015–2016

UITBREIDBARE WEB-GEBASEERDE
INTERMODALE ROUTEPLANNER VOOR BELGIË

Brecht VAN DE VYVERE

Promotors:

Prof. dr. ir. R. Van de Walle

Prof. dr. ir. E. Mannens

Begeleiders:

dr. ir. R. Verborgh

ing. P. Colpaert

Masterproef voorgedragen tot het behalen van de graad van
INDUSTRIEEL INGENIEUR: INFORMATICA

Copyright 2003, 2004, 2005, 2006, 2007, 2008 Gaspard Lequeux. Alle rechten voorbehouden. Dit werk mag worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm, geluidsband, kleitabletten, elektronische of welke andere wijze ook, onder de volgende voorwaarden:

- Vermelding van de auteur.
- Niet commercieel. Dit vloeit voort uit het feit dat sommige paragrafen overgenomen werden uit ?, wiens werk niet mag gebruikt worden voor commerciële doeleinden.
- Als je dit werk wijzigt en/of verdeelt, moet dit gebeuren onder dezelfde voorwaarden.

De auteur is niet aansprakelijk voor eventuele fouten en onvolledigheden in dit werk.

Versie 0.6.0

Gecompileerd op: 12 september 2015

Woord vooraf

Deze cursus werd in November 2003 voor het eerst uitgegeven in het kader van een Zeus (Studenten Werkgroep Informatica) initiatief om thesistudenten te helpen hun thesis in een aantrekkelijke vorm te gieten. De tekst is opgevat als een soort minithesis zodat studenten gemakkelijk alles kunnen overnemen. Om die reden zijn de bronbestanden beschikbaar gemaakt op het internet.¹

Hoewel sommige deeltjes van deze cursus expliciet gericht zijn op het maken van een thesis, kan de tekst gebruikt worden als algemene inleiding op L^AT_EX. In 2004 zag ook Prof. Ottoy dit in. Vandaar dat deze cursus nu deel uitmaakt van de lessen informatica in de tweede bachelor bio-ir van de Universiteit Gent. Bij die gelegenheid heeft hij de cursus volledig nagekeken op taalfouten, waarvoor dank.

In 2006 heeft Prof. Dawyndt de cursus ook grondig herlezen en nagekeken, waarvoor dank. Sindsdien wordt deze cursus gebruikt in de lessen Computergebruik gegeven in de eerste bachelor Informatica van de Universiteit Gent.

*Voor het schrijven van deze handleiding L^AT_EX, werd gebruik gemaakt van twee werken: *A guide to L^AT_EX* van ? en *Handleiding L^AT_EX* van Piet van Oostrum (1996). Uit dit laatste werden zelfs hele paragrafen overgenomen. Daarnaast werd natuurlijk rijkelijk geput uit de documentatie die meegeleverd wordt met L^AT_EX zelf.*

Minder belangrijke delen worden in een kleiner lettertype weergegeven. Verder zijn er verschillende voetnoten die verwijzen naar L^AT_EX documentatie op een Debian GNU/Linux systeem. Niemand gebruikt dat natuurlijk. Maar

¹<http://zeus.ugent.be/~gaspard/latex>

de bedoeling ervan is dat de lezer weet dat de documentatie bestaat, welke bestandsnaam die heeft en waar ongeveer die te vinden is in de \LaTeX directorystructuur. Het is dan niet zo moeilijk meer om in je favoriete besturingssysteem te zoeken naar de desbetreffende bestandsnaam.

In een standaard MikTeX installatie (dé \LaTeX distributie voor Windows), zijn de helpfiles van de verschillende \LaTeX packages te vinden in `C:\texmf\doc\latex`. In `C:\texmf\doc\guides` zijn enkele algemene handleidingen te vinden over \LaTeX .

Het woord vooraf dient ook om mensen te bedanken:

- De mensen van Zeus, voor de stimulerende Vrije–Open Source sfeer en het organiseren van lessen hier rond.
- Schamper, het studentenblad van de Universiteit Gent, voor het leveren van de promotor van dit werk.
- Rudy Gevaert, die in het academiejaar 2001-2002 als eerste een \LaTeX les gaf.
- Geert Vernaeye, voor het eerste contact met \LaTeX en het C-voorbeeld op bladzijde ??.
- Mensen die fouten rapporteerden en/of verbeteringen suggereerden: David De Wolf, Annelies Huyck, Yves Nevelsteen, Stijn Gors, Geert Vernaeye, Michiel Meire, Hendrik Maryns, Olivier Verhoogen, Jean-Pierre Ottoy, Hugo Coolens, Lieven Clement, Andy Peene, Reinout Debergh, Frederik De Schrijver, David van der Ha, Brecht Donckels, Dominique Lebbe, Christopher De Dobbelaere, Paul Vogels, Heidi Vanparys, Stijn Depuydt, Veerle Gevaert, Joke Van Hevele, Katrien De Dauw, Francis Santens, Nicolas Vanden Bossche en Peter Dawyndt.
- De (thesis)studenten van het Boerenkot, die in 2003 gevraagd hebben naar deze handleiding.

\LaTeX lijkt in het begin moeilijk: alles in tekstmode, geen knopjes, je moet speciale commando's kennen om iets te bereiken ... De eerste dagen zul je inderdaad enkele problemen ondervinden. Zoeken, doorbijten en hulp vragen aan meer ervaren gebruikers zullen ervoor zorgen dat je na enkele weken zelfs je wiskundige redeneringen rechtstreeks in \LaTeX uitvoert.

Deze handleiding is waarschijnlijk niet foutloos. Het rapporteren van meer dan één fout, zorgt voor je naam in de volgende editie van dit woord vooraf. Ook inhoudelijke opmerkingen zijn steeds welkom.

*Gaspard Lequeux
Gent 9 Augustus 2006*

Inhoudsopgave

1	Inleiding	1
1.1	Hoe het begon	1
1.1.1	TEX	1
1.1.2	L ^A TEX	1
1.2	Het verwerken van tekst	2
1.3	Voor- en nadelen	4
1.4	Het L ^A TEX compileerproces	5
1.5	L ^A TEX installatie en gebruik onder Windows	6
2	Conclusies en perspectieven	8

Hoofdstuk 1

Inleiding

1.1 Hoe het begon

1.1.1 \TeX

\TeX (uitspraak “Tech” als in technéut, kan ook als ‘TeX’ geschreven worden) is een computer-programma van Donald E. ? . Het is speciaal ontworpen voor het zetten en drukken van wiskundige teksten en formules. Een belangrijk kenmerk van \TeX is dat je er zelf nieuwe functies in kan programmeren (zogenaamde macro’s). Het pakket is dus uitbreidbaar. Vele mensen die uitbreidingen geschreven hebben, maken die publiek beschikbaar. Het is dus aangewezen om eerst te kijken wat er al bestaat, vooraleer zelf iets te implementeren.

1.1.2 \LaTeX

\LaTeX (uitspraak “Lah-tech”, kan ook als ‘LaTeX’ geschreven worden) is een zogenaamd macro-pakket dat door Leslie ? werd geschreven en gebruik maakt van \TeX . Het stelt de auteur in staat zijn publicaties op eenvoudige wijze en met gebruik van een op voorhand opgegeven structuur, met boekdrukwaliteit te zetten en af te drukken.

Om die speciale zes \LaTeX letters te krijgen, moet je het commando `\LaTeX\` gebruiken. \TeX verkrijg je met het commando `\TeX\`.

1.2 Het verwerken van tekst

Het maken van printbare documenten kan op verschillende manieren gebeuren. Enerzijds heb je de wuziwuk¹ tekstverwerkers, waar het merendeel van de mensen mee vertrouwd is, en anderzijds heb je de zogenaamde *markup* tekstprocessing programma's, zoals L^AT_EX.

De wuziwuk tekstverwerkers zijn initieel gemakkelijk in gebruik. Je typt en er verschijnt juist hetzelfde als hetgeen op papier zal komen.² Voor korte documenten is dit zeer gemakkelijk. Maar voor lange documenten werkt dit niet meer zo goed: die drie enters die je op het einde van een bladzijde had gezet om de daarop volgende titel op een nieuw blad te krijgen (omdat dat mooi was) blijken na het doorvoeren van een verbetering plots in het midden van een blad te staan. Of de figuren die zo mooi geschikt waren, beginnen rond te zweven, of nog erger: worden vervangen door een rood kruis. Zelfs als dat je bespaard wordt, blijft er nog altijd het probleem van consistentie: bij de ene figuur heeft het bijschrift lettergrootte 10, bij de andere 11.

Vandaar de andere manier: L^AT_EX. De opmaak wordt gescheiden van de inhoud. De auteur geeft de structuur aan en de computer maakt aan de hand van de gestructureerde tekst een prettig leesbaar document. Dit kan een printbaar bestand zijn (met `latex`) of een webpagina (met `latex2html`). Met één commando kan je je thesis publiceren op het *www*!

De tekst wordt getypt in een gewone editor. In een editor kun je alleen tekst typen: geen knopjes om vet of cursief te zetten, geen animaties, geen figuren invoegen. Enkel tekst, platte tekst. Dit lijkt archaïsch, omdat de meeste mensen alleen *wordpad* als editor kennen, maar er bestaan veel betere: `vim` is er één van, `emacs` is een andere. Beiden zijn vrij verkrijgbaar op het net. Onder Windows bestaat er zoiets als *winedit* (te betalen) maar ook `vim` kan daar geïnstalleerd worden. Dé L^AT_EX-editor onder Windows is echter TeXnicCenter.³

Platte tekst. Da's niet veel. Hoe krijg je dan titeltjes, paginanummering, opmaak? Wel, er bestaan van die regels om goede teksten te maken. Die kun je vanbuiten leren en dan toepassen in een wuziwuk programma. Maar

¹Wat U Ziet Is Wat U Krijgt

²Althans, dit is de theorie. In de praktijk durft dit wel eens lelijk tegen te vallen.

³www.texniccenter.org

die regeltjes toepassen kan je ook laten doen door de computer. In \LaTeX zeg je: dit is een hoofdstuk, dit een titel, dit is een belangrijk woord, dit is letterlijke tekst, in de buurt van deze paragraaf moet deze figuur komen, dit is een Latijnse naam voor een bacterie. Allemaal in de teksteditor (hoe je dat doet, zien we later). Je tekstbestand geef je dan te eten aan \LaTeX . Op basis van het tekstdocument dat je gemaakt hebt, maakt \LaTeX een mooi printbaar document. Waar je zegt: dit is de titel van een nieuw hoofdstuk, gaat \LaTeX een nieuwe bladzijde beginnen, “Hoofdstuk x” schrijven (waarbij x het juiste nummer van het hoofdstuk is), op een nieuwe lijn gaan staan en dan je hoofdstuk titel in een gepast lettertype plaatsen.

Hoe geef je je wensen te kennen aan \LaTeX ? Via speciale tekstsequenties natuurlijk. Aanduiden dat een nieuw hoofdstuk begint, gaat als volgt:

```
\chapter{Een nieuw hoofdstuk}
```

Die *backslash* maakt de \LaTeX compiler duidelijk dat wat volgt geen gewone tekst is, maar een commando dat betrekking heeft op hetgeen tussen de accolades staat.

Iets over opmaakontwerp

Typografisch ontwerpen is een vak, dat men moet (kan) leren. Ongeoefende auteurs maken vaak zware typografische fouten. Vele leken denken ten onrechte dat typografie alleen maar een kwestie van smaak is; wanneer een document er mooi uit ziet, is het ook goed ontworpen. Daar documenten echter gelezen dienen te worden, zijn leesbaarheid en begrijpbaarheid belangrijker dan het uiterlijk.

Voorbeelden: De lettergrootte en nummering van titels van hoofdstukken en paragrafen moet zo gekozen worden, dat de structuur van hoofdstukken en alinea's duidelijk herkenbaar is. De regellengte dient zo gekozen te worden, dat vermoeiende oogbewegingen voor de lezer voorkomen worden en niet zo, dat het papier zo mooi mogelijk ‘gevuld’ wordt met letters.

Met wuziwuk-pakketten maken auteurs soms mooie, maar slecht leesbare documenten. Een veel gemaakte fout is bijvoorbeeld het gebruiken van te veel verschillende lettertypes of het onderlijnen van titeltjes — een typografische ramp. \LaTeX voorkomt typografische fouten, omdat het de auteur

dwingt de logische structuur van een tekst aan te geven. \LaTeX gebruikt dan automatisch de meest geschikte opmaak.

1.3 Voor- en nadelen

\LaTeX heeft, vergeleken met andere tekstverwerkingspakketten, de volgende voordelen:

- De gebruiker hoeft maar een paar, gemakkelijk te begrijpen, commando's te leren. Deze commando's betreffen alleen de logische structuur van het document, de gebruiker hoeft zich nauwelijks bezig te houden met de technische details.
- Complexe structuren zoals voetnoten, literatuuropgaven, inhoudsopgaven, tabellen, formules etc. en zelfs eenvoudige tekeningen, kunnen zonder al te veel moeilijkheden gemaakt worden. Het zetten van complexe wiskundige formules is bijzonder gemakkelijk in \LaTeX .
- Het geproduceerde document is consistent: alle titels zien er gelijkaardig uit (zelfde lettertype, zelfde letterstijl), de bijschriften bij tabellen en figuren hebben allemaal dezelfde opmaak en het citeren gebeurt altijd op dezelfde manier (inderdaad, wie houdt er zich in deze tijd van krachtige computers nog bezig met het handmatig opstellen van een referentielijst?).
- Het is beschikbaar voor bijna alle computersystemen, en gratis.
- \LaTeX is zeer stabiel. Crashes of corrupte bestanden zijn ongekend. \LaTeX bronbestanden bevatten eigenlijk gewone tekst. Zij nemen dan ook weinig plaats in en kunnen gemakkelijk gebackupt worden op diskette.
- Vele mensen gebruiken \LaTeX . Er bestaan dus een hele hoop pakketten (extensies) voor \LaTeX die zeer gespecialiseerde dingen doen. Op www.ctan.org⁴ zijn alle mogelijke pakketten verzameld. De meesten hiervan zijn waarschijnlijk echter al op je systeem geïnstalleerd.

\LaTeX heeft natuurlijk ook nadelen:

⁴Comprehensive \TeX Archive Network

- Binnen de door \LaTeX ondersteunde opmaakstijlen kunnen weliswaar enige parameters gevarieerd worden, maar ingrijpende afwijkingen van de gebruikte opmaakstijl kunnen slechts met veel moeite tot stand gebracht worden.
- Het intypen van \LaTeX commando's is ingewikkelder dan het aanbrengen van opmaak met moderne tekstverwerkers, zeker als die voorzien zijn van een menu-besturing. Ook al omdat het resultaat van de commando's niet direct zichtbaar is.

Kort gezegd zijn voor kleine documenten (brieven, posters, memo's) tekstverwerkers meestal in het voordeel, terwijl voor grote, complexe documenten \LaTeX vaak betere resultaten en uiteindelijk minder werk oplevert.

1.4 Het \LaTeX compileerproces

\LaTeX leest dus een tekstbestand in, en aan de hand daarvan worden een aantal bestanden gegenereerd waarvan vooral één bestand ons interesseert, namelijk het printbaar document. Het verwerken van een \LaTeX bronbestand door de \LaTeX compiler gebeurt met het commando `latex thesis.tex`, waarbij `thesis.tex` het \LaTeX bestand is dat we willen compileren.

Oorspronkelijk was het document dat uit de latex-compiler kwam, een **dvi**-bestand (van *DeVice Independent*). Dit bestand is onafhankelijk van een printer. Om het naar een printer te kunnen sturen, moet het met een zogenaamde *printer driver* worden omgezet naar een bestand geschikt voor die bepaalde printer. Een veel gebruikt printerformaat is het **ps** of PostScript formaat. Het nieuwere formaat om printklare bestanden op te slaan en door te geven is **pdf** (*Portable Document Format*).

Het `latex` commando genereert een *dvi* bestand. Dit kan worden omgezet naar een *ps* bestand met het commando `dvips thesis.dvi -o thesis.ps` (of korter: `dvips thesis -o`). Voor het omzetten van een *dvi* bestand naar een *pdf*-bestand bestaan er twee conversieprogramma's: `dvipdf` en `dvipdfm`. De syntax is iets anders dan bij `dvips`, namelijk: `dvipdf thesis.dvi` en er wordt een `thesis.pdf` aangemaakt.

Maar het is beter om rechtstreeks met *pdf* te werken. Met `pdflatex` maak je rechtstreeks een *pdf*-bestand aan. De syntax is dezelfde als voor `latex`:

```
pdflatex thesis.tex
```

Dit document werd gecompileerd met `pdflatex`. We zullen verder ook alles uitleggen voor `pdflatex`. Dus wanneer gesproken wordt over het laten compileren door de latex compiler, bedoelen we eigenlijk de `pdflatex` compiler.

Naast een dvi of pdf bestand, genereert \LaTeX nog twee (of soms meer) andere bestanden: een **aux** en een **log**-bestand. In het *log*-bestand komen alle commentaren, die je op het scherm ziet voorbij rollen. Het *aux*-bestand is een help-bestand voor \LaTeX zelf. Als je bijvoorbeeld in hoofdstuk 1 verwijst (verwijzingen worden behandeld in sectie ??) naar een bladzijde in hoofdstuk 2, kan \LaTeX tijdens het verwerken van hoofdstuk 1 niet weten welke bladzijde dit is. Wanneer echter hoofdstuk 2 wordt verwerkt, gaat \LaTeX in de aux-file de juiste bladzijde bijhouden en de volgende keer dat je de latex-compiler loslaat op je document, zal de juiste bladzijde ingevoerd worden. Zie sectie ?? op bladzijde ?? voor meer uitleg. Het is om die reden dat een document finaal minstens twee keer door de latex-compiler moet gejaagd worden (maar we zullen in de loop van deze cursus zien dat het tot vier keer nodig kan zijn).

Voor de mensen die met Windows werken en bang zijn van een *command line*: geen nood. TeXnicCenter, een goede editor voor \LaTeX onder Windows, beschikt over een aantal knopjes waar je kan op klikken en die dan al deze zaken voor jou uitvoeren.

1.5 \LaTeX installatie en gebruik onder Windows

Onder GNU/Linux wordt \LaTeX bij de meeste distributies standaard meegeleverd (het zogenaamde texlive pakket). Onder Windows is het iets moeilijker.

- Dé \LaTeX distributie voor Windows is MikTeX.⁵ Op deze website staat uitgelegd wat we moeten doen om \LaTeX te installeren op onze computer. Na de installatie hebben we ‘slechts’ een \LaTeX compiler, geen teksteditor, noch een handig knopje waarop we kunnen klikken om ons *tex*-bestand om te zetten naar een *pdf*-bestand. Slechts de DOS

⁵www.miktex.org

command line is beschikbaar en *notepad* voor het bewerken van onze *tex*-bestanden.

- Om het resultaat te bekijken, moeten we (althans voor pdf_latex), een pdf-lezer installeren. De beste hiervoor is Acrobat Reader van Adobe.⁶ Meestal is die echter al geïnstalleerd.
- Om plezier te werken, installeren we TeXnicCenter.⁷ Dit is een geïntegreerde omgeving voor het maken van L^AT_EX documenten. Je hebt van die handige menu's waarin je kan kiezen uit verschillende commando's, zodat je ze niet allemaal moet onthouden. Het is best om eerst Acrobat Reader te installeren, zodat TeXnicCenter hiermee kan linken tijdens de installatie.

Omdat L^AT_EX gebaseerd is op redelijk oude computercode, kan de latex-compiler moeilijk of niet omgaan met spaties (en andere rare tekens) in bestandsnamen of directories. Installeer MikTeX dus niet in *C:\Program Files* maar bijvoorbeeld wel in *C:\miktex*. Gebruik ook nooit spaties in je *tex*-bestandsnamen. Dus NIET opslaan in *C:\Mijn Documenten* maar wel in *C:\latexbestanden* of iets dergelijks. Dit zal veel kopzorgen vermijden.

⁶www.adobe.com

⁷www.texniccenter.org

Hoofdstuk 2

Conclusies en perspectieven

Elk goed werk eindigt met algemene slotconclusies (pleonasme) en een blik op wat beter kan (maar niet elk werk dat daarmee eindigt is een goed werk). Nu, bij deze cursus zijn er niet veel conclusies. Perspectieven zijn er des te meer. We hebben het vooral gehad over de syntax van \LaTeX . Niet over de programma's er rond. Mensen die met \LaTeX blijven werken, worden door de jaren heen zeer kritisch over de kwaliteit van hun documenten. Maar niet alleen hun documenten, heel hun (computer) tot op het pathologische af.¹ Bijgevolg zijn de programma's er rond zeer belangrijk.

- **make** Bij de finalisatie van een document, moeten verschillende commando's opgeroepen worden: 1 keer `latex`, 1 keer `bibtex`, 3 keer `latex`, 1 keer `makeindex`, 2 keer `latex`. Met een *Makefile* kan je dat allemaal inkorten tot één enkel commando: `make full`.
- **ispell** Spellingscontrole.
- **xfig** Voor maximale homogeniteit tussen de figuren en de tekst. De letters op je figuur zijn dezelfde als die in je document.
- **vim** De teksteditor. In het begin moeilijk te gebruiken, maar één keer je er mee weg bent, lijken alle andere tekstverwerkers zo kleurloos.
- **gimp** *GNU Image Manipulation Program*. Om figuren op te poetsen.
- **gnuplot**, **grace** Voor het omzetten van data in grafieken.

¹Waarom denk je, werd deze cursus geschreven op een Debian GNU/Linux systeem? Omdat dat gratis is? Nee! Omdat het beter is? Speelt vast en zeker mee. Maar de belangrijkste reden is de VRIJHEID van dat systeem!