

# Don't Panic

*"In nice and relaxing letters"*

Har dette hæfte på forsiden, simpelthen fordi du ikke bør gå i panik. Lidt ligesom "The hitchikers guide to the galaxy", som bestemt er anbefalet læsning, selvom det nok ikke er skrevet i

---

## L<sup>A</sup>T<sub>E</sub>X

---

En kort introduktion af L<sup>A</sup>T<sub>E</sub>X for de virkelig hungrende, og alle dem der bare gerne vil lære det.

*Digital typesetting means patterns of 0s and 1s, and computer science can be thought of as the study of 0s and 1s. Therefore, it dawned on me for the first time, that I, as a computer scientist, would be able to help solve the printing problem that was worrying me so much. I didn't need to know about metallurgy or optics or chemistry or anything scary like that; all I had to do was to construct the right pattern of 0s and 1s. Therefore it was almost an obligation for a computer scientist like myself to study the problem carefully.*

Don E. Knuth  
T<sub>E</sub>X påbegyndt 5. maj 1977  
Første version klar 1979 på Stanford  
T<sub>E</sub>X 3.0 klar 1989 (den nuværende version)

## Indhold

0.1	Hvad er L <sup>A</sup> T <sub>E</sub> X, gør det ondt? . . . . .	1
<b>1</b>	<b>Strukturen i L<sup>A</sup>T<sub>E</sub>X</b>	<b>1</b>
1.1	Input? . . . . .	1
1.2	Filosofien bag det hele . . . . .	1
<b>2</b>	<b>Dokumentopbygning</b>	<b>3</b>
2.1	Den spæde start . . . . .	3
2.2	Pakker og udvidelser . . . . .	4
2.3	Tekst, sådan generelt . . . . .	4
2.3.1	Ting du ikke må skrive . . . . .	5
2.4	Tekst struktur . . . . .	6
2.5	Kom SÅ IGANG . . . . .	7
<b>3</b>	<b>Dybere ind i mørket</b>	<b>8</b>
3.1	Miljøer . . . . .	8
3.2	Float miljøet . . . . .	9
3.2.1	Figurer og <i>includegraphics</i> . . . . .	10
3.2.2	Tabeller og <i>tabular</i> . . . . .	11
3.3	Referencer . . . . .	13
3.3.1	Eksempel med referencer . . . . .	13
3.3.2	Struktur i dine labels . . . . .	14
3.4	Grafik . . . . .	14
<b>4</b>	<b>Matematik</b>	<b>15</b>
4.1	Det grundlæggende . . . . .	15
4.2	Opsætning af matematik . . . . .	16
<b>5</b>	<b>Stoerre rapporter</b>	<b>18</b>
5.1	Struktur i store dokumenter . . . . .	18
5.2	At lave en kildeliste . . . . .	19
5.3	Custom Kommandoer . . . . .	19
<b>6</b>	<b>Installation af L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b>	<b>21</b>
6.0.1	Drevstruktur . . . . .	21
6.1	Installation af MiK <sub>T</sub> E <sub>X</sub> . . . . .	21
6.1.1	At compile uden shell . . . . .	22
6.1.2	Dansk orddeling med MiK <sub>T</sub> E <sub>X</sub> . . . . .	23
6.2	Installation af GsView og GhostScript . . . . .	24
6.3	Installation af T <sub>E</sub> XnicCenter . . . . .	24
6.3.1	Dansk stavekontrol i T <sub>E</sub> XnicCenter . . . . .	24
<b>7</b>	<b>Hvorfor L<sup>A</sup>T<sub>E</sub>X ?</b>	<b>25</b>
<b>8</b>	<b>Kildeliste/Links</b>	<b>26</b>

## Figurer

1	L <sup>A</sup> T <sub>E</sub> X cyklus . . . . .	1
2	Et dokument, klar til at skrive i . . . . .	5
3	Niveau-delning, af dokument . . . . .	7
4	Figurskabelon . . . . .	11
5	Matematik eksempel . . . . .	17
6	Et typisk hoveddokument. Saadan burde det ihvertfald se ud . . . . .	18
7	En struktur til stoere rapporter . . . . .	18
8	Et system til kildelister . . . . .	19
9	Custom kommandoer . . . . .	20

## Tabeller

1	Tabelskabelon . . . . .	12
---	-------------------------	----

### Resumé

Du sidder nu med din egen danske introduktion til L<sup>A</sup>T<sub>E</sub>X. På et eller andet tidspunkt bør du dog læse "The not so short guide to L<sup>A</sup>T<sub>E</sub>X"<sup>1</sup>, hvilket er den måde, vi andre lærte det på dengang i tidernes morgen.

Den nemmeste måde at lære L<sup>A</sup>T<sub>E</sub>X på er ved at bruge det. Derfor anbefales det at have sin computer parat, så eksemplerne kan følges. Men lad det ikke blive ved at skrive eksemplerne af, leg med tingene, og ændr på parametrene/opsætningen. Det kan jo ikke gå værre end galt.

Gennem hele guiden har jeg antaget, at du bruger TeXnicCenter som "shell" (se side 1). Desuden håber jeg også for dig, at du bruger M\$ Windows ©, for det gør jeg<sup>2</sup>.

Der er også et afsnit om at installere L<sup>A</sup>T<sub>E</sub>X, dette er placeret bagerst (da det ikke har noget med indlæringen at gøre). En installation kommer man dog ikke uden om, så gør det nu, så eksemplerne her giver mere mening.

Det er dog vigtigt at huske, at dette IKKE er den definitive guide til svaret paa livets spoergsmaal<sup>3</sup>. Derfor er der sikkert ogsaa en million andre maader at goere tingene paa, hvilket jeg synes man skal goere.

Idet jeg d. 13.09.03 er begyndt paa uni, og samtidig fik lyst til at skrive noget mere, skal det lige bemærkes at jeg ikke bruger æ ø og å længere, idet jeg er gaaet helt over til DVORAK tastaturet. Det er altsaa ikke en fjel i L<sup>A</sup>T<sub>E</sub>X, men en bevidst handling fra min side... Deal with it.

God fornøjelse

Jes Toft Kristensen  
dinmor [at] hotmail.com  
17. oktober 2003

*ADVARSEL! Dette er skrevet under kraftig indflydelse af pebernødder, makrelmadder og pink floyd, hvilket nogen steder har medført en absurd mængde sort og indforstået humor. So don't say you weren't warned!*

---

<sup>1</sup><ftp://ctan.unsw.edu.au/tex-archive/info/lshort/english/lshort.pdf>

<sup>2</sup>Ikke for at fornærme bestemte parter, der måske ville have andre præferencer end jeg (rent operativsystemsmæssigt). Mit udsagn skal kun tages som en påpegning af, at denne "manuals" dele, specielt delen om installation, tager udgangspunkt i M\$ Windows ©, hvorved proceduren, for eksempelvis en installation kan være anderledes, enddog alternativ, på andre systemer.

<sup>3</sup>Se i stedet "Hitchhikers guide to the galaxy, af Douglas Adams , det er et must læsning!!!

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*

## 0.1 Hvad er L<sup>A</sup>T<sub>E</sub>X, gør det ondt?

Nej, kun lidt i starten. Og kun fordi man skal lære det.

Det er et typesetter system, der kan det samme som word, amipro, word-perfect og alle de andre teksteditorer. Denne teksteditor gør dog det hele på en anden måde. Men det skal du nok få lært.

# 1 Strukturen i L<sup>A</sup>T<sub>E</sub>X

Selve systemet er grundlæggende opbygget af en compiler. Denne compiler tager din kildekode (.tex filer) og laver den om til et .pdf dokument. På figur 1 er systemet vist. Dette består grundlæggende af 3 stadier. Først laver man sit input. Så giver man det til compileren, og så får man output. Altså sker disse ting ikke på samme tid, hvilket er noget anderledes, end hvad PC brugere normalt er vant til.



Figur 1: L<sup>A</sup>T<sub>E</sub>X cyklus

Den compiler du højst sandsynligt bruger er MiKTeX. Derudover kan man installere en shell, så din kode får andre farver osv. Du læser dette dokument, hvilket tyder på at du også har et program til at læse .pdf dokumenter med.

## 1.1 Input? Kan det spises, og giver det så output?

Det, du fodrer din compiler med er .tex filer. Dette er egentligt bare "ren tekst" (ASCII), som du kan skrive på alle computere. Du skal bare skrive i notepad eller en anden shell<sup>4</sup> (TeXnicCenter, Emacs eller notepad eksempelvis). Hvor du så skal sørge for, at endelsen på filen er .tex (omdøb den!)

Din kildekode (herefter tex-filer) består af nogen kommandoer, som kompilatoren (MiKTeX) så kan "oversætte" til .pdf, hvilket er rart idet pdf ikke fylder særligt meget, og er et gratis format (L<sup>A</sup>T<sub>E</sub>X er også gratis, det der stod i licensen, som du nok ikke læste, var, at MiKTeX er gratis). Det er nemmest at forstaa, hvis man sammenligner med html, for det er i princippet det samme.

## 1.2 Filosofien bag det hele

Ideen i L<sup>A</sup>T<sub>E</sub>X er, at man ikke selv skal sætte dokumentet op rent typografisk. Det eneste, du skal bekymre dig om er at skrive indholdet og så holde en god struktur i dokumentet.

<sup>4</sup>en shell er bare et program til at skrive i

Dette lyder måske ikke som en stor ændring, men det er det! Idet du skriver dit dokument i ASCII kode, kan du ikke se det endelige output (.pdf) direkte<sup>5</sup>. Så glem alt, hvad du hidtil har lært af unoder i *M\$ word*© og andre WYSIWYG<sup>6</sup> programmer. Layout er der nogen, der har gjort for dig, og de er meget bedre til det, end du er. Så la' vær med at prøve at pille!

Som et eksempel er denne side angivet i ren .tex nedenfor.

```
introduktion.tex (uddrag) ->
*****
\subsection{Filosofien bag det hele}
Ideen i \LaTeX{} er, at man ikke selv skal sætte dokumentet op,
rent typografisk. Det eneste du skal bekymre dig om, er at
skrive indholdet, og så holde en god struktur i dokumentet.
```

Dette lyder måske ikke som en stor ændring, men det er det! Idet du skriver dit dokument i ASCII kode, kan du ikke se det endelige output (.pdf) direkte\footnote{Med erfaring får man dog et ret godt billede af hvordan tingene ser ud}. Så glem alt hvad du hidtil har lært, af unoder i \emph{M\$ word}\copyright{} og andre WYSIWYG\footnote{What You See Is What You Get} programmer. Layout er der nogen der har gjort for dig, og de er meget bedre til det end du er. Så la' vær med at prøve at pille!

\\ \\

Som et eksempel er denne side angivet i ren tekst ovenfor.

```
*****
```

Alt dette skal vi selvfølgelig nok gennemgå senere. Men prøv at læse det igennem, for at se om du kan gætte nogen ting. (faktisk kan man, med lidt erfaring, gætte kommandoer, tænk lidt over hvad \underline{tekst}gør).

---

<sup>5</sup>Med erfaring får man dog et ret godt billede af hvordan tingene ser ud

<sup>6</sup>What You See Is What You Get

## 2 Dokumentopbygning, NU! Jeg vil igang

Så lad os komme igang. Fyr op for TeXnicCenter, åbn/lav et nyt projekt under File – > new project. Vælg et tomt projekt, og giv det et eller andet navn, jeg har kaldt mit for ”l0rn”. Sæt rullegardinet<sup>7</sup> fra ”LaTeX => DVI” til ”LaTeX => PDF”. Sådan, så får vi pdf ud.

### 2.1 Den spæde start

Du skulle så gerne have en hel blank side. Det første vi gør, er at fortælle L<sup>A</sup>T<sub>E</sub>X compileren, hvad vi skriver. Dette gøres ved at skrive

```
*****
\documentclass[a4paper,10pt]{report}
*****
```

på den første linie i dokumentet. Denne kommando består af 2 dele. Den første del er dens ”options” (som er [a4paper,10pt] ). Her siger vi at vi bruger a4 papir (i stedet for amerikansk letter), og at skriftstørrelsen i dokumentet er 10 ”point”. Der står flere options angivet i lshort (”The not so short guide to L<sup>A</sup>T<sub>E</sub>X”).

Den næste del, som kommandoen består af, er dens ”definition”. Dette er omkranset af { og } og siger, at dette dokument er en rapport.

Det næste, vi skal, er at påbegynde vores dokument. Dette gør vi simpelthen ved at skrive

```
*****
\begin{document}
*****
```

desuden skal vores dokument også slutte igen, som vi gør med

```
*****
\end{document}
*****
```

så... Imellem begin og end kan vi så skrive vores tekst. Prøv at skrive, så hele din tex fil ser sådan ud:

```
*****
\documentclass[a4paper,10pt]{report}
\begin{document}
Stort er smukt, men det kommer nok senere.
\end{document}
*****
```

så kan du prøve at ramme F7 tasten på dit keyboard, hvilket skulle få TeXnicCenter til at compile dit dokument. Bagefter trykker du på F5, hvorefter dit pdf view program skulle komme frem, med vores smukke dokument<sup>8</sup>.

Du har nu lavet dit første dokument. Giv dig selv et klap på skulderen, og gå videre...

<sup>7</sup>det var dog et frygteligt navn for en ting

<sup>8</sup>Hvis det ikke gør, så råb om hjælp, for det burde det

## 2.2 Pakker og udvidelser

L<sup>A</sup>T<sub>E</sub>X er i sig selv bare en samling makroer, som ligger frit fremme. Dvs. at alle kan bruge dem frit og gratis. Dette har så affødt, at folk skriver deres egne udvidelser til L<sup>A</sup>T<sub>E</sub>X, mange af dem gør et godt stykke arbejde, så det vil vi da udnytte.

Denne udvidelse ligger i form af pakker, og du kan endda skrive din egen. Med MiKTeX fulgte der en mængde pakker, som vi egentligt bare skal gøre aktive, for at de virker. Dette gøres i vores ”preamble”, som er det stykke, der er imellem *documentclass* og *begin{ document}* . I dette stykke må der ikke stå normal tekst, da din preamble er forbeholdt kommandoer.

Når vi skal have en pakke ind, skriver vi simpelthen

```
*****
\usepackage[options]{pakke}
*****
```

eksempelvis vil vi gerne kunne bruge billeder i vores dokument. Dette gøres muligt ved

```
*****
\usepackage[pdftex]{graphicx}
*****
```

hvor optionen ”pdftex” fortæller programmet, at vi compiler (næsten)direkte til pdf, og at selve pakken hedder ”graphicx” (så kan du jo prøve at gætte på, hvad den gør).

En anden ting, der er rart, er danske bogstaver. L<sup>A</sup>T<sub>E</sub>X kan ikke skrive æøå som udgangspunkt, så vi skal udvide vores skrifttype med de fonte der hedder ”latin1”.

```
*****
\usepackage[latin1]{inputenc}
*****
```

derudover skal den også dele vores ord, som om de var danske og skrive overskrifter og indholdsfortegnelser, som om de var danske

```
*****
\usepackage[danish]{babel}
\captionsdanish
*****
```

Altså skulle vi nu have et dokument, der ser ud som på figur 2.

## 2.3 Tekst, sådan generelt

Tekst i L<sup>A</sup>T<sub>E</sub>X skriver du bare ind, som om det var normal tekst. Dog er der nogen specielle (nemme) regler for lineskift og specialtegn.

L<sup>A</sup>T<sub>E</sub>X opfatter et ”normalt” lineskift som ingenting. Hvis du vil tvinge L<sup>A</sup>T<sub>E</sub>X til at lave et lineskift, skal du bruge dobbelte backslash

```
*****
Her skal der være \\ et lineskift.
*****
```

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*



---

```

\documentclass[a4paper,10pt]{report}
\usepackage[pdftex]{graphicx}
\usepackage[latin1]{inputenc}
\usepackage[danish]{babel}
\captionsdanish
\begin{document}
Stort er smukt, men det kommer nok senere. Nu med ÆØÅ og dansk orddeling
\end{document}

```

---

Figur 2: Et dokument, klar til at skrive i

ellers finder L<sup>A</sup>T<sub>E</sub>X selv ud af, hvornår ord skal deles, og linierne skal skiftes. Prøv at skrive en lang tekst (5 - 6 liniers random ord) i TeXnicCenter, og compile det. Du vil nu se, at L<sup>A</sup>T<sub>E</sub>X selv har delt ordene, så marginen passer.

Inde i dine random ord fra før kan du prøve at lave et dobbelt linieskift, ligesom i

```
*****
```

```
axe abe test sten papir
linieskift? Hvor blev det af?
```

```
videre med tingene, for det her giver ingen mening
```

```
*****
```

og se så, at der er opstået et nyt, indrykket afsnit. Et godt råd er at starte et nyt afsnit, hver gang en ny tanke i din tekst begynder (men det kan din dansklærer helt sikkert fortælle dig mere om end jeg kan).

Der er en sidste opdeling af tekst, som er afsnittet. Dette bryder linien helt, så der opstår en meget kraftig opdeling. Brug dette til at strukturere dit dokument yderligere. Et afsnit laves med 2 gange dobbelt backslash:

```
*****
```

```
axe abe test sten papir
\\ \\
videre med tingene, for det her giver ingen mening
```

```
*****
```

### 2.3.1 Ting du ikke må skrive

Idet makroerne bygger på ren tekst, er der desværre nogen tegn, som ikke kan bruges direkte. Det vigtigste er, at du ikke kan bruge backslash (en saadan: \ fætter her). Idet alt er skrevet i ren tekst, leder compileren efter backslashes,

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*

og naar den finder et, ved den at der kommer en kommando bagefter. Det er derfor at alle kommandoer i L<sup>A</sup>T<sub>E</sub>X er skrevet med et backslash foran.

Nedenunder er der skrevet et backslash som kode, dog med dollartegn udenom. Dette er fordi at backslash bliver opfattet som et matematik-tegn, og DERFOR skal der dollartegn udenom<sup>9</sup>.

Desuden kan du ikke bruge normale "gåseøjne" ( " ), men skal bruge dobbelt ( ' ) i stedet. Altså skal ting i gåseøjne på følgende måde<sup>10</sup>

```
*****
Dette ''skal i gåseøjne''.
her er et $\backslashbackslash{ }$
*****
```

En anden ting er procent-tegnet, som man heller ikke må strø om sig med gavmild hånd. Et procenttegn i tex-filen, vil få compileren til at opfatte alt det der står efter, som et notat... Det vil sige at det ikke kommer med i outputtet. Der er dog en anden måde at skrive det på, nemlig ved at bruge et backslash før procent-tegnet

```
*****
her er et notat %som består i dette, hele linien ud.
men dette er et \% -tegn.
*****
```

Desuden er der en mængde andre specialtegn, som man aldrig bruger alligevel. De står i lshort.

## 2.4 Struktur eller, hvor er den "overskrift 1" knap

L<sup>A</sup>T<sub>E</sub>X har en stærk mekanisme til at strukturere dokumenter. Faktisk er den så skarp, at du nok bliver nødt til at tænke over det, du vil skrive, før du begynder på det. Dette er ikke kun et nedrigt plot fra din dansklærer, men faktisk et godt råd.

Det grundlæggende niveau i L<sup>A</sup>T<sub>E</sub>X er "section". Under dette er der 3 niveauer, nemlig "subsection", "subsubsection" og "subsubsubsection". Over section er der også 2 niveauer, nemlig "chapter" og "part". Kapitel og del (chapter og part) deler dokumentet meget kraftigt, og er kun til at dele rapporter op med (eksempelvis teoridele, forsøgsbeskrivelse osv.).

Det hele fungerer ved, at man skriver

```
*****
\section[Det her står i indexet]{Det her står i selve teksten}
*****
```

Denne underdels overskrift er skrevet:

```
*****
\subsection[Tekst struktur]{Struktur eller, hvor er den ''overskrift 1'' knap}
*****
```

I dette dokument er der kun brugt sections, subsections, subsubsections og subsubsubsections. På figure 3 er der vist, hvordan inddelingen er opdelt.

<sup>9</sup>Dette skal jeg nok forklare senere.. hehe

<sup>10</sup>Der er en anden, og mere korrekt maade at goere dette paa, jeg gider bare lige ikke finde den

```
\part{Det stor forsøg}
\chapter{Det med larvefødder}
\section{Om aben og dens udvikling}
\subsection{Forsøgsresultater}
\subsubsection{Usikkerheder!}
\subsubsubsection{Damn, det er lavt, og kommer ikke med i indexet}
```

---

Figur 3: Niveau-delning, af dokument

Nummereringen af sektioner, afdelinger osv. sker helt automatisk. Hvis du vil have et index i dit dokument, skriver du bare

```
*****
\tableofcontents      % indeholdsfortegnelse
\listoffigures        % figurindex
\listoftables         % tabelindex
*****
```

hvilket giver dig index over indholdet, figurer og tabeller. Det er ikke et krav, at de alle er med, eller at du overhovedet bruger det.

For at få indexerne rigtige skal du compile 2 gange. Således at L<sup>A</sup>T<sub>E</sub>X kan hente de rigtige referencer ind <sup>11</sup>.

## 2.5 Kom SÅ IGANG

Du har nu lært nok, til at du kan skrive danske stile og andet godt... Derfor anbefaler jeg dig nu at lege lidt rundt med det. Hvis der genereres uanede mængder af fejl, så slet din nyeste ændring og prøv igen. Der kan jo ikke gå noget galt.

---

<sup>11</sup>Den fulde undskyldning står i lshort

### 3 Flashy ting, med punktopstillinger og sager

Idet du er nået hertil, er du vel nok standhaftig. Det tegner godt!! Du skal nok blive en god T<sub>E</sub>Xnician engang.

#### 3.1 Miljøer

Det, vi nu skal møde, er ”environments” (miljøer). Fælles for dem er, at de gør ting ved noget, der er i dem (hvilket miljøer jo har en tendens til). Et miljø startes og slutes med

```
\begin{environment}
miljøet ville gøre noget ved denne tekst
\end{environment}
```

Tilbage er der kun at vise nogen af de forskellige miljøer, som det er muligt at bruge. Vi kan starte med en punktopstilling

- Den første ting i listen
- Den næste ting
- skovl
  - et underniveau
  - næste punkt i et underniveau
- saks
- bagepapir
- heri kan der også stå uanede mængder af tekst desuden også med lineskift og se hvor pænt det ser ud

denne ser sådan ud, når man læser koden

```
*****
\begin{itemize}
\item Den første ting i listen
\item Den næste ting
\item skovl
\begin{itemize}
\item et underniveau
\item næste punkt i et underniveau
\end{itemize}
\item saks
\item bagepapir
\item heri kan der også stå uanede mængder af tekst
desuden også med lineskift og se hvor pænt det ser ud
\end{itemize}
*****
```

Et andet miljø der virker på denne måde (med ”item”) er enumerate. Dette skriver dog (selv) numrer op ud for tingene.

```
*****
\begin{enumerate}
  \item Den ting der får nummer 1
  \item den ting der ikke får nummer 3
  \item prøv at bytte om, for at teste nummereringen.
\end{enumerate}
*****
```

Tada.. og de klappede i deres små hænder

1. Den ting der får nummer 1
2. den ting der ikke får nummer 3
3. prøv at bytte om, for at teste nummereringen.

Derudover kan man sætte sin tekst op til at være, højre, venstre eller center placeret. Dette gøres med

```
*****
\begin{flushleft}
Enhedslisten (spild af tid)
\end{flushleft}
```

```
\begin{center}
Spild af tid
\end{center}
```

```
\begin{flushright}
Det yderste højre
\end{flushright}
*****
```

giver :

Enhedslisten (spild af tid)

Spild af tid

højre

### 3.2 Float, som i "they all float, down here"

Hvis man vil have større plamager (figurer og tabeller) i sit dokument, har L<sup>A</sup>T<sub>E</sub>X en funktion, der hedder "float" environments.

Ideen i L<sup>A</sup>T<sub>E</sub>X er, at man ikke selv skal rode med layout. Dette gælder altså også for figurer og tabeller. L<sup>A</sup>T<sub>E</sub>X lader alle "figurer" og "tabeller" flyde til et passende sted, så det er typografisk forsvarligt (pænt). Dette medfører i starten lidt frustration, men man vænner sig til det. Du kan nemlig ikke længere skrive "på ovenstående figur/tabel" idet du ikke ved hvor din figur er (!). I stedet for bruger man referencer, som så også virker konsekvent. Se side 13 for nærmere info om dette (accepter bare at `\label{ }` er der).

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*

Man indsætter en figur, som om det var et miljø, altså med `\begin{ }`. Derefter giver man miljøet nogen options der fortæller L<sup>A</sup>T<sub>E</sub>X, hvor du synes den må placere floaten (figuren/tabellen). Disse options er

- h - for "placer den lige her, hvis du synes det ser pænt ud"
- t - for "placer den i toppen, hvis du synes det ser pænt ud"
- b - for "placer den i bunden, hvis du synes det ser pænt ud"
- p - for "find en random side til sidst, og smid det der, hvis du bliver tændt af det"
- ! - for "den skal ihvertfald være i denne sektion (skrives efter de andre options)"

Man kan også give disse options på samme tid, hvorpå L<sup>A</sup>T<sub>E</sub>X vælger den der passer bedst (ser pænest ud).

Efter at man har begyndt sit miljø, kan man egentligt putte det i det, som man har lyst til. Dette kan være alt fra matematik og billeder, til autogenerated tegninger, eller en dansk stil (altså, man kan putte alt i det). L<sup>A</sup>T<sub>E</sub>X Bestemmer så hvilken størrelse det har/får, og placerer det så.

Som alt i livet er der også her en dualitet. L<sup>A</sup>T<sub>E</sub>X har indbygget 2 typer float miljøer. Den ene hedder "figure", imens det andet hedder "table". Den eneste forskel er dog, at table kan spænde over flere sider, og bliver opført i indexet over tabeller (`\indexoftables`, se side 2.4). Figure miljøet bliver derimod opført i indexet over figurer (meget smart).

### 3.2.1 Figure - med billeder

Den nemmeste syntaks at forstå er den for et typisk billede. Så den er på figur 4.

1. Miljøet startes, og vi angiver placeringsoptions
2. Vi fortæller at i resten af miljøet skal det hele centreres.
3. Her indsætter vi grafikken. Se side 14. Tilsæt selv indhold her.
4. Caption fortæller hvad der skal stå under figuren. Det i firkantede parenteser kommer til at stå i indexet, det i krøllede parenteser kommer til at stå i under figuren.
5. Så giver vi figuren en intern reference. Se side 13.
6. Miljøet afsluttes.

Her kan man altså fjerne 2 og 3, og erstatte det med indhold efter eget valg. Caption og label bør man dog altid have med, medmindre man selvfølgelig ikke vil have noget navn eller nogen reference til figuren. Husk altid at skrive caption før label.

Der er en udvidelse til float miljøet, der bl.a. gør det muligt at tvinge dine figurer til at "SÆT DEN HER, FORDI JEG SIGER DET", med en H option

---

```

\begin{figure}[htb!]                               %1
  \center                                          %2
  \includegraphics[scale=1,angle=0]{bild/billede.jpg} %3
  \caption[Figurskabelon]{Typisk billed/figurindsættelses kode} %4
  \label{fig:figureX}                             %5
\end{figure}                                       %6

```

---

Figur 4: Typisk billed/figurindsættelses kode

til placeringen. Meget handy når man skal lave folde-ud diagrammer (for det skal man!). Pakken hedder *float* (meget kreativt). En anden pakke, der kan placere dine figurer så teksten ”wrapper” rundt om dem er enten *wrapfigur* eller *floatflt* (floatflt er nemmest at bruge). Se på det allestedsnærværende CTAN.

### 3.2.2 Tabeller - med tabeller

Den anden type float der findes er table floaten. Selve floaten er der ikke noget hokus pokus i (jeg har ihvertfald ikke fundet nogen forskel fra figure floaten, udover det med indexet. Men man kan jo altid blive klogere). Det interessante her er et ”undermiljø” hvor man kan lave smukke tabeller.

Dette miljø hedder tabular, hvor float miljøet hedder table. Syntaksen kan godt være lidt kompliceret først, men det lærer man senere. Første starter man miljøet, dernæst angiver man om kolonnerne skal venstre, center eller højreorienteres, dette gøres med l, c eller r (left, center og right). Desuden angiver man hvor mange kolonner der skal være, og om der skal være adskillende streger imellem kolonnerne.

Dernæst begynder man at skrive på tabellen. Hver rubrik adskiller man med &-tegnet, og man starter en ny linie med \\ (dobbelt backslash). Man indsætter en horisontal linie, ved at skrive *\hline*.

Dette er lavet på tabel 1. Det der sker her er at

1. Vi begynder det flydende table miljø, og angiver hvor vi synes L<sup>A</sup>T<sub>E</sub>X skal placere tabellen henne.
2. Så begynder vi tabular miljøet, der kan lave selve tabellen. Vi siger at der er 5 kolonner (5 bogstaver), og imellem nogen af dem, skal der være lodrette streger. Desuden laver vi en horisontal linie, med *\hline*.
3. Så begynder det sjove. Vi skriver ind, og adskiller med &-tegnet. Jeg har skrevet nogen små kommandoer, som er indbygget i latex. Desuden laver vi dobbeltunderstregning her, med to gange *\hline*.
4. Vi hælder vand ud af ørene.

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*

5. Og vi bliver ved.
6. og ved ...
7. duracell (og dog, vi afslutter lige tabular miljøet)
8. Så får den lige en caption, som normalt.
9. og en label (reference).
10. Så afsluttes float miljøet.

---

```

\begin{table}[htb] %1
\begin{tabular}{|l|cr|cc|} \hline %2
\today      & \LaTeX{} & \LaTeXe{} & \TeX{} & sager  \\ \hline %3
dato        & abe      & \ldots    & sager  & noget  \\ %4
enandendag  & slappe af & ingenting & indhold? & sager  \\ \hline %5
$2+2$      & matematik & random    & modnar  & fluxtæthed \\ %6
\end{tabular} %7
\caption[Tabelskabelon]{Typisk tidsplan, omend lidt kort} %8
\label{tab:tabelX} %9
\end{table} %10

```

---

17. oktober 2003	L <sup>A</sup> T <sub>E</sub> X	L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub>	T <sub>E</sub> X	sager
dato	abe	...	sager	noget
enandendag	slappe af	ingenting	indhold?	sager
2 + 2	matematik	random	modnar	fluxtæthed

---

Tabel 1: Typisk tidsplan, omend lidt kort

Som med enhver anden ting, findes der et trick der gør dit liv lidt nemmere. Et godt råd er, at stille det hele op i et regneark (OpenOffice Calc, eller MS Excel<sup>®</sup>), og så kopiere det over i .tex dokumentet. Dette er måske snyd, men det er nemmere, og fylder ikke nær så meget som et excel dokument. Desuden ser det pænere ud. Du skal ikke bekymre dig om bredden af kolonnerne, det finder L<sup>A</sup>T<sub>E</sub>X selv ud af.

Som udgangspunkt kan man ikke have fodnoter i en tabel. Til dette skal du bruge udvidelsespakken *tabularx*, som er at finde på CTAN.



### 3.3 Referencer

Referencesystemet i L<sup>A</sup>T<sub>E</sub>X er meget simpelt. Det fungerer ved at man giver "labels" (som på computerdisketter), som man så kan refererer til. Alt kan som udgangspunkt lables. Det man så kan kalde tilbage er tingens "nummer". Dette kan være beregningsnummer, sektionsnummer, figurnummer etc. plus sidetal. Men det forstår man nemmest med et lille eksempel<sup>12</sup>.

```
*****
\subsubsection{Eksempel med referencer}
\label{sektioneksempel}
\begin{eqnarray}
a^2 + b^2 & = & c^2 \label{beregning1} \\
E & = & m \cdot c^2 \label{labelskanvaerealt}
\end{eqnarray}
Hvis vi betragter udregning ( \ref{beregning1} ), på side
\pageref{beregning1}, ser vi at den indeholder en ligegyldig
formel, samt at den ikke har noget at gøre med
( \ref{labelskanvaerealt} ). Dog er begge i sektion
\ref{sektioneksempel}.
```

En anden ting man kan betragte er figur \ref{fig:figureX} på side \pageref{fig:figureX}. Denne har jo fået en label i sig (inde i figur-miljøet), som den så angiver, i stedet for sektionsnummeret.

```
*****
```

---

#### 3.3.1 Eksempel med referencer

$$a^2 + b^2 = c^2 \tag{1}$$

$$E = m \cdot c^2 \tag{2}$$

Hvis vi betragter udregning ( 1 ), på side 13, ser vi at den indeholder en ligegyldig formel, samt at den ikke har noget at gøre med ( 2 ). Dog er begge i sektion 3.3.1.

En anden ting man kan betragte er figur 4 på side 11. Denne har jo fået en label i sig (inde i figur-miljøet), som den så angiver, i stedet for sektionsnummeret.

---

<sup>12</sup>for det er jo dødsimpelt

Efter at have noerdet lidt med dette er jeg blevet opmaerksom paa pakken "varioref", som tilfoejer "paa naeste side" og "paa forrige side", alt efter hvor objektet er placeret. Man starter dette med `\vref{noget}` i sit dokument, og kalder den med `usepackage[danish]{varioref}`.

### 3.3.2 Struktur i dine labels

Det næste er ikke et krav, men en god ting. Idet man må bruge kolon i sine labels, kan man opbygge en god struktur, simpelthen ved at sætte nogen indekser på. Eksemplet er at kalde sine figurer "fig:navn#". Udregninger kan kaldes "eq:ting#" osv. Således har man altid lidt styr over, hvad der er hvor.

Hvis i er flere om at skrive en rapport er det også en god ide at lægge sine initialer ind også. Så er man sikker på at man ikke får nogen ens referencer, som jo ikke må forekomme.

## 3.4 Grafik

L<sup>A</sup>T<sub>E</sub>X kan ogsaa haandtere grafik. Dette goeres foerst og fremmest ved at aktivere pakken "graphicx"<sup>13</sup>. Som option skal man bruge den compiler man anvender, alstaa typisk "pdftex".

Naar man har startet pakken, kan L<sup>A</sup>T<sub>E</sub>X haandtere formater af foelgende type: .jpg .png .tif, hvor man dog boer holde sig til jpg og png, idet tif (hvis du da nogensinde har hoert om dette) godt kan vaere lidt random. Desuden skal man vaere lidt forsigtig med alt for store billeder, idet de kan vaere svaere for L<sup>A</sup>T<sub>E</sub>X at haandtere.

Den typisk maade at faa grafik/figurer med i et dokument, er ved at putte dem i et figure environmet. Kommandoen for at indsaette grafik ser saadan ud:

```
*****
\includegraphics[scale=1.0,angle=0]{bild/billede.jpg}
*****
```

hvor includegraphpics' options er scale (skaler billede op og ned) og angle, hvor angle angiver rotationen paa billedet (rotation mod uret, vaerdien er i grader). I selve parametrene paa includegraphics angiver man hvor billedet er. Her er jeg gaaet ud fra at billedet ligger i underbiblioteket "bild", set relativt fra der hvor projektets hovedfil ligger<sup>14</sup> men mere om strukturen i dokumenter senere.

Hvis du mangler nogen programmer til at tegne/haandtere grafik i, saa kan du altid kigge paa [www.gimp.org](http://www.gimp.org) (der er ogsaa en windows "port", udgave). Desuden er [www.xnview.org](http://www.xnview.org) ogsaa et supergodt og gratis program, der kan et samme, og lidt til, som ACDSec.

<sup>13</sup>Der findes en simplere version, nemlig "graphics", men der er jo ingen grund til at indskraenke sig

<sup>14</sup>(

## 4 Metamitak på den rå måde

Grrrrr... kender i det ikke godt? Man har lige stillet sig over til pisrenden på tog-stationen i Århus. Idet man ikke kan urinere med en hånd, har man stillet sine tasker lige inden om hjørnet. Problemet er så, at jeg får den skøre tanke, hvad nu hvis en eller anden smart idiot rendte med dem, imens man står der? Hvad gør man så? Idet dette er en relativt sandsynlig (hvis tyve tænkte mere!) situation, vil jeg på det kraftigste opfordre dig til at have en løsning. Ellers kan det gå dig ilde.

Men det har jo ikke noget med L<sup>A</sup>T<sub>E</sub>X at gøre, så nu vil jeg straks blive seriøs igen, og forklare hvordan matematikken hænger sammen, for det er virkelig L<sup>A</sup>T<sub>E</sub>X stærke side.

### 4.1 Det grundlæggende

Matematik skrives i miljøer. Der findes flere forskellige miljøer man kan bruge, her med deres specielle egenskaber.

Det simpleste er `"displaymath"`, som kun kan vise en linie, uden nummerring. Stillet op, vil

```
*****
\begin{displaymath}
2 + \frac{2\pi}{4^2} = n_i^3
\end{displaymath}
*****
```

give

$$2 + \frac{2\pi}{4^2} = n_i^3$$

(og sådan er det hele vejen igennem) hvor `\frac{ tæller }{ nævner }` giver en brøk. Et underscore, som efter n, vil sænke skriften til et index. Samtidig vil den smukke hat hæve skriften som en potens.

Hvis du har set andre fancy rapporter, ved du også at matematikken kan have sådanne neate tal ude i siden. Til dette kan du bruge "equation" miljøet. Denne kan dog heller ikke være i flere linier.

```
*****
\begin{equation}
\sum_5^d f(x) + t = \alpha \cdot \Omega^{\{3 + 1\}}
\end{equation}
*****
```

giver

$$\sum_5^d f(x) + t = \alpha \cdot \Omega^{3+1} \quad (3)$$

Her giver `\sum` summeringstegnet. Dette er en forberedt streng (automatiseret kommando), som der er mange af. Der findes et fuldt skema i lshort, men jeg vil bruge de mest normale her.

Det mest normale miljø at bruge, er "eqnarray". Fordelen ved dette er, at det

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*

kan have flere linier, at der er reference tal, og at det kan opstilles med & -tegn, til at opstille et array. Men lad os tage det et trin af gangen.

De flere linier opnåes, ved at sætte et dobbelt backslash efter.

```
*****
\begin{eqnarray}
f(t) = \int_n^i 3,5 \cdot x^2 dx \\
\Updownarrow \\
\frac{f(t)}{x^2} = \int_n^i 3,5 dx
\end{eqnarray}
*****
```

giver (som er en ugyldig reduktion!)

$$f(t) = \int_n^i 3,5 \cdot x^2 dx \quad (4)$$

$$\Updownarrow \quad (5)$$

$$\frac{f(t)}{x^2} = \int_n^i 3,5 dx \quad (6)$$

At opløfte og sænke er ikke noget krav, jeg gør det bare fordi man kan. Desuden er der en lille ting, med at ophæve eller indexere. Hvis man skal have opløftet/sænket mere end et tegn, skal det omslutes af tuborgklammer. Som her

```
*****
\begin{eqnarray}
\int_{\text{danmark}}^{\text{indien}} \text{gris} = n/a
\end{eqnarray}
*****
```

$$\int_{\text{danmark}}^{\text{indien}} \text{gris} = n/a \quad (7)$$

## 4.2 Opsætning af matematik

Naar man har lidt styr paa det grundlaeggende, kan man begynde at overveje at faa det til at se paent ud. Det foerste man skal laere er at bruge det rigtige environment. I det foelgende vil jeg gaa ud fra eksemplet i figur 5.

I den foerste linie initialiserer man eqnarray, som er en af de math environments der understoetter nye linier, og som har stykkenummerering. I linie 2 skriver vi et helt normalt stykke op. Dog saetter vi & udenom = tegnet. Saaledes at vi i naeste linie (linie 3) ogsaa kan saette & -tegn ind, hvorved vi opnaar, at = tegnene saettes over hinanden. I linie 4 saetter vi qquad og quad ind for at give afstand imellem de forskellige tegn. Normalt styres afstanden mellem tegn helt automatisk, men paa denne maade kan man altsaa selv styre afstanden.

Hvis man skal have puttet matematik ind i en tekst, som eksempelvis i  $\sin(\theta) = \pi$ , saetter man dollartegn udenom. Dette er en lille hurtig kommando, der ikke starter paa et miljøe.

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*

\*\*\*\*\*

eksempelvis i  $\sin(\theta) = \pi$ , sætter man dollartegn

\*\*\*\*\*

Hvis man bliver virkeligt involveret i matematikken, har "American Mathematics Society" skrevet en udvidelsespakke, ved navn "AMS-math" eller lignende, som det helt sikkert er værd at kigge paa.

\*\*\*\*\*

```
\begin{eqnarray}
4+4 & = & 2345 \cdot x & \backslash & \%1
34 + 43 + 45 + 34 & = & \sin(5x) \cdot \cos(5y) & \backslash & \%2
a \quad b \quad = & 5 & & & \%3
\end{eqnarray}
*****
```

som bliver til:

$$4 + 4 = 2345 \cdot x \quad (8)$$

$$34 + 43 + 45 + 34 = \sin(5x) \cdot \cos(5y) \quad (9)$$

$$a \quad b \quad 5 \quad (10)$$

Figur 5: Matematik eksempel

## 5 Weeee... vi skal skrive 80 sider

Naar man kommer videre med sit liv, og engang faar lyst til at skrive store rapporter, er det vigtigt at have en god struktur i sine dokumenter. Typisk vil man ogsaa vaere flere om at skrive en rapport, hvorfor man er tvunget til at have tingene liggende i mindre filer (ellers kan man ikke skrive paa samme tid, hvilket er lidt spild af tid).

### 5.1 Struktur i store dokumenter

Alt dette bygges grundlaeggende op med "input" og "include" kommandoerne. Man har et hoveddokument, som man saa smider andre tekst-stykker ind i. Et saadant hoveddokument kan se ud som paa figur 6.

---

```

\documentclass[a4paper,10pt]{article} %start legen
\input{preamble}    %alle dine pakker og custom kommandoer
\input{headers}    %configurer dine headers her

\begin{document}   %start med at skrive
\input{intro}      %indsaet forside, synopsis, tableofcontents osv.
\include{kapitel1} %foerste del, kan godt vaere \part{} eller \chapter{}
\include{kapitel2} %anden del, kan godt vaere \part{} eller \chapter{}
\include{kapitel3} %tredie del, kan godt vaere \part{} eller \chapter{}
\include{kilder}   %smid kilder ind her

\end{document}     %saa er den ged barberet....

```

---

Figur 6: Et typisk hoveddokument. Saadan burde det ihvertfald se ud

I disse kapitelX-filer kan man saa inputte underdelene. Det er en god ide at bygge op med en biblioteksstruktur, hvilket goer det hele endnu nemmere at finde rundt i. Paa figur 7 er der vist en anbefalelsesvaerdig opdeling, men den kan jo altid goeres finere.



Figur 7: En struktur til stoere rapporter

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*

Til sidst er det vigtigt at forstaa forskellen paa "input" og "include". For forskellen er nemlig vild og voldsom(!) include kommandoen er "den stoerste". Naar man inkluder noget skiftes der automatisk til en ny side, og man kan ikke bruge en include i noget der er inputtet (men man kan godt bruge en input i en input, eller i en include).

Hvis man derimod bruger en input, er det det samme, som at tage det der staar i filen, og smide det direkte ind. Eksempelvis kan man bruge en input i en preamble.

## 5.2 At lave en kildeliste

Er man saa heldig at finde kilder til sine store ting, skal det jo selvoelgelig ogsaa med. Den nemmeste maade er at lave en `begin{enumerate}`, og saa have sine kilder som items under dette. Tricket er saa at kaste en label paa, som man saa referere til. Princippet er vist paa figur 8.

---

```

*****
\begin{enumerate}
  \item Om BOFH, the Bastard Operator From Hell \label{kilde:bastard} \\\
    http://bofh.ntk.net/Bastard.html - En virkelig sej samling historier (...)
  \item \label{kilde:sager1d}
    sager.1d.dk Et godt sted at skaffe denne guide
\end{enumerate}
*****
Se kilde \ref{kilde:bastard} for yderligere forklaring. Eller
kilde \ref{kilde:sager1d} for mere hjaelp.

```

---

Figur 8: Et system til kildelister

L<sup>A</sup>T<sub>E</sub>X er nemlig saa smart, at den selv giver nummeret ud, paa det der refereres til. Hvis du taenker over det, skaelner du jo heller ikke mellem figurer, sider du giver en label eller tabeller.

Den helt rigtige maade at goere det paa er vha. udvidelsen `bibTEX`, men det er et helt kapitel for sig selv, og et udmaerket emne for selvstudie.

## 5.3 Custom Kommandoer

Hvis man en dag faar lyst til at skrive sine egne kommandoer, kan dette ogsaa lade sig goere. Her bruger man blot en indbygget funktion, som man putter i sin preamble. Dette er vist paa figur 9.

Syntaksen for `newcommand` er foerst at angive hvad kommandoen skal hede. Saa hvor mange options den har, og til sidst selve kommandoen. Argumenter

*Don't panic  
it's only L<sup>A</sup>T<sub>E</sub>X  
17. oktober 2003*

fra options delen kan saa hives frem med `#x`, hvor `x` er optionens nummer (fra 1 til `n`).

I de to foerste eksempler laver vi en nem genvej, saa vi ikke skal skrive saa meget, naar vi indfoere vores matematikopgave. Her er der ikke options paa, idet vi blot skal starte og slutte et miljøe.

Det sidste eksempel er maaske en lidt nytteles funktion. Vi definerer en kommando der goer vores tekst skraa (emphasized). Vi hiver simpelthen vores argument ind i en emphasize box, hvilket er en rar ting.

---

```
*****
\newcommand{\bmath}[0]{\begin{eqnarray}}
\newcommand{\emath}[0]{\end{eqnarray}}

\newcommand{\skraa}[1]{ \emph{#1} }
*****
\bmath
2 + 2 = 5
\emath
\skraa{Denne tekst vil vaere skraa}
```

---

Figur 9: Custom kommandoer



## 6 Installation af L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>

L<sup>A</sup>T<sub>E</sub>X består jo som sagt af flere forskellige komponenter. Alt efter hvordan man anvender det, skal man bruge mellem 2 og 4 programmer.

Den nemmeste måde, hvor man compiler direkte ud i .pdf, skal man bruge 2, måske 3 programmer til. Disse programmer er

- En L<sup>A</sup>T<sub>E</sub>X compiler (MikT<sub>E</sub>X)
- En shell (T<sub>E</sub>XnicCenter)
- En viewer (Adobe Acrobat eller GsView)

De programmer, jeg vil gennemgå installationen af, er MikT<sub>E</sub>X, T<sub>E</sub>XnicCenter og GsView. Disse programmer er de (efter min mening) nemmeste at bruge, og i min uvidenhed om alternativerne har jeg valgt at bruge disse. Desuden er Adobe Acrobat installeret på de fleste systemer.

### 6.0.1 Drevstruktur, for SATAN!<sup>15</sup>

Hvis du senere skal have hjælp til dit system (ting går jo af og til galt), skal du sørge for at have en fornuftig drev-struktur. Dette indebærer, at man har alle ting samlet et sted, så de er nemme at overskue for eventuelt servicepersonel.

Min befaling er derfor at oprette en afdeling i din program mappe, der simpelthen hedder `c:\programmer\latex\` (eksempelvis). Derefter skal du pine død også sørge for, at programfilerne lægges derunder. Dette er altså der, hvor du skal have alle dine L<sup>A</sup>T<sub>E</sub>X ting liggende.

### 6.1 Installation af MiK<sub>T</sub>E<sub>X</sub>

Den største ting, og selve L<sup>A</sup>T<sub>E</sub>X, er miktex compileren. Denne kan hentes enten fra [www.miktex.org](http://www.miktex.org) hvor den er gratis, eller også kan den skaffes på CD.

Hvis man downloader den, henter man første en lille "wizard". Dette er et lille program, der gør tingene utroligt meget nemmere. Dette program hedder, meget kreativt, `setup.exe`. Når du har hentet denne lille sag, så kører du den, hvorefter den spørger dig, om du vil installere fra et "local package repository" eller downloade en installation. Her vælger du først at downloade en installation.

I den næste menu vælger du en "large" installation. Heri er der det mest nødvendige, og denne kan dække alles behov. En large installation indeholder de mest normale pakker, dog uden at fylde utroligt meget<sup>16</sup>. I den næste menu (efter at have valgt din pakkestørrelse) skal du vælge et mirror. Dette er simpelthen en server, der har filerne liggende. Personligt plejer jeg at bruge den danske ([ftp.sunsite.dk](http://ftp.sunsite.dk)) eller den belgiske (som jeg ikke kan huske adressen på), men dette er i princippet ligegyldigt, da de bare skal sende pakkerne til dig.

Efter dette vælger du et download dir. Dette bliver dit local package repository, som du skal installere fra. Når installationen er færdig, kan du slette filerne igen. Brug eksempelvis `c:\install\`. Læn dig derefter tilbage, og nyd at hente dit gratis program.

<sup>15</sup>Antipædagogisk, men korrekt

<sup>16</sup>Hvis du har blod på tanden, kan du også hente en "small", og så senere spæde den op med eksterne komponenter

MikT<sub>E</sub>X kan også fåes på CD. Her vil pakkerne typisk ligge i underbiblioteker, som du så lægger ind. Led på CDen efter noget dokumentation om, hvor pakkerne skal kopieres ind. Problemet er, at de skal ligge rigtigt, ellers kan installations-wizarden ikke finde dem. Hvis det fejler totalt med at finde dokumentationen på CDen, kan du til nød læse i log-filen (download-”herstændato”.log), hvori der står noget ala

```
download-(dato).log ->
```

```
-----
```

```
MiKTeX setup wizard 2.2.7 Report
```

```
Date: (dato)
```

```
Time: (tidspunkt)
```

```
Setup path: \?\?\setup.exe
```

```
starting download...
```

```
Local folder: (her står dit local package repository)
```

```
-----
```

Hvor linen ”Local folder” angiver hvor du skal lægge filerne fra CDen hen for at kunne installere.

Efter at du har placeret filerne korrekt, kører du din wizard igen (setup.exe), og vælger denne gang at installere MiK<sub>T</sub>E<sub>X</sub>. Dernæst bliver du spurgt, om du vil have et shared eller private environment. Spørg din systemansvarlige om alle, eller kun dig skal have adgang til L<sup>A</sup>T<sub>E</sub>X. Hvis du er i tvivl, så vælg et private environment. Herefter vælger du din installation folder og **husk så den drevstruktur!** Det, du så bliver spurgt om, er dit local TEXMF tree, hvor du igen skriver et bibliotek, der passer ind i drevstrukturen. Læg din localtexmf folder i samme niveau som din MiK<sub>T</sub>E<sub>X</sub> installation. Herefter er det bare at bekræfte det hele, og så begynder installationen.

### 6.1.1 At compile, uden shell (kun for nørder! men meget sundt)

Når du så har installeret MiK<sub>T</sub>E<sub>X</sub>, kan du også teste den, simpelthen ved at compile. Skriv en lille fil, måske den famøse

```
foo.tex ->
```

```
-----
```

```
\documentclass{article}
```

```
\begin{document}
```

```
hello world!
```

```
\end{document}
```

```
-----
```

Hvilket man kan compile, ved at bruge sin konsol (start i MS-dos, hav et program der har en kommand-line, eksempelvis totalcommander, eller gøre køre- > og skrive ”cmd”).

I din konsol navigerer du så til der, hvor du gemte din tex-fil<sup>17</sup>. Herefter skriver du

<sup>17</sup>I Dos, ”CD biblioteksnavn” for ”Change Dir biblioteksnavn”

```
latex foo.tex
```

hvorefter den skal lave en fil, der hedder foo.dvi . Herefter kan du skrive

```
dvips foo.dvi (hvorefter den laver .ps (postscript))
```

eller

```
dvipdfm foo.dvi (hvorefter den laver .pdf (portable-document-file))
```

Hvis dette ikke virker, skal du have lagt MiK<sub>T</sub>E<sub>X</sub> installationen i din system-path (rolig nu! **Don't Panic.**) Få fat i den nærmeste nørd, og sig, at du skal have lagt

```
... \Latex\miktex\miktex\bin\*.*
```

i din system path<sup>18</sup>. Efter dette burde du kunne compile uden problemer.

### 6.1.2 Dansk orddeling med MiK<sub>T</sub>E<sub>X</sub>

Når du engang har skrevet dit dokument, og har fået den til at bruge det herlige danske sprog, så får du måske højest sandsynligt og ret sikkert en fejlbesked af typen

```
warning! no hyphenation patterns loaded for danish
defaulting to .....
```

Dette er tegn på, at du ikke har installeret orddelingsmodulen for dansk. Dette er imidlertid en smal sag, som vi da lige kan gennemgå her.

- find din MiK<sub>T</sub>E<sub>X</sub> installation.
- gå til `\source\generic\babel\`
- åbn filen `language.dat` med notesblok eller anden editor (IKKE WORD!)
- tilføj linien `"danish dkhyph.tex"` (uden anførselstegn)
- i din start menu, går du til `start - > programmer - > miktex - > miktex options`
- under fanen `"languages"` sætter du et flueben ud for `danish`
- under fanen `"general"` trykker du på knappen `Update now`, i underafdelingen om `format files`.
- den laver nu nye `"format"`-filer hvorefter det hele burde virke. Ellers lok en L<sup>A</sup>T<sub>E</sub>X nørd til at hjælpe dig<sup>19</sup>.

<sup>18</sup>Lok eventuelt med MCDonalds, pizza eller sex

<sup>19</sup>Lok eventuelt med RAM moduler, webcams eller sex

## 6.2 Installation af GsView og GhostScript

Programmet GsView er en viewer, der kan læse både postscript (.ps) og .pdf. GhostScript er makropakke som GsView bruger til at læse filer med. Installer først GhostScript og derefter GsView.

Fordelen ved disse programmer er, at de ikke åbne dokumenter med "exclusive rights". Hvis et program (som eksempelvis Adobe Acrobat) åbner et dokument, kan ingen andre skrive i det. Dette betyder at du skal lukke Acrobat, opdatere dokumentet (compile det) og så åbne det igen i Acrobat. Med GsView skal du ikke bøvle med dette, den opdatere bare automatisk. Dette forstår du måske ikke lige med det samme, men det skal du nok opdage en dag... hehe

## 6.3 Installation af T<sub>E</sub>XnicCenter

T<sub>E</sub>XnicCenter er en shell, som gør livet nemmere for dig. Blandt andet kan den compile uden at du skal skrive mærkelige ting i din konsol. Derudover har den også menuer med alle de græske bogstaver og andet godt, så du ikke skal slå dem op hele tiden, når du skal skrive dem.

Men det hele startes med at køre .exe filen, som sikkert hedder "TXCSetup\_1Beta6\_01.exe" (eller noget).

Installationen burde ikke være noget problem, men bare husk at holde en ordentlig drevstruktur.

Når du starter T<sub>E</sub>XnicCenter, skal den konfigureres. Dette består hovedsageligt i, at bekræfte den i, at du har en MiK<sub>T</sub>E<sub>X</sub> installation og så pege på filen yap.exe. Den nemmeste måde at finde den på er ved at søge på strengen "yap.exe"<sup>20</sup>. Desuden skal du nok også finde din .ps viewer (GsView.exe) og din .pdf viewer (GsView32.exe anbefalet, ellers acrobat.exe).

Udover dette vil du også blive bedt om et work dir. Dette kan være alt (C:\work), og betyder ikke så meget.

Så burde din shell være klar. Du kan nu compile ved at trykke F7 (hvis du var klog og startede det hele som projekt), eller ctrl + F7 (hvis du bare brugte file - > new). Husk at indstille T<sub>E</sub>XnicCenter til at compile som *LaTeX* => *PDF*, i stedet for *LaTeX* => *DVI*.

### 6.3.1 Dansk stavekontrol i T<sub>E</sub>XnicCenter

Der understøttes også dansk stavekontrol. Dette kræver dog at du henter danske ordbøger. Disse findes på [www.OpenOffice.org](http://www.OpenOffice.org) (eller i forbindelse med Mozilla browseren). Til dette henter du 2 filer ned. De hedder sikkert

- da\_K.dic
- da\_Dk.aff

disso to skal ind i mappen

```
... \TEXnicCenter\language\..
```

hvorefter du genstarter T<sub>E</sub>XnicCenter, vælger Tools - > options - > spelling og sætter sproget til da.

<sup>20</sup>yap = Yet Another Previewer

## 7 Hvorfor? (fordi jeg siger det!)

Her skulle der saa vaere en laengere argumentation for at bruge L<sup>A</sup>T<sub>E</sub>X, men det gider jeg ikke nu. Idet du er naaet hertil, saa er du nok hooked, og argumenterne har haft deres tur. Saa afinstaller du bare waardhhhh, det skal du ikke bruge laengere (jo, men hvis du skal have en office-pakke, saa brug i det mindste den gratis, OpenOffice.org).

Hvis du derimod mangler lidt ammunition til at lukke munden paa alle M\$ drengene, saa er der indtil flere fortrinlige artikler paa nettet. Eksempelvis "The evils of WYSIWYG" (What You See Is What You Get) og "The virtues of ASCII" eller lignende.<sup>21</sup>

Men ellers kan du jo altid paapege hvor sjovt de der 60 siders waard rapporter er...og hvis skruen skal have en tand mere, kan du smide nogen billeder ind, saa har waard ihvertfald tabt.

---

<sup>21</sup>Er ikke sikker paa at det er de korrekte titler, men de er ikke svaere at finde

## 8 Kildeliste/Links

Her er den dokumentation, jeg har brugt, da jeg skrev dette dokument. Desuden er der her også de vigtigste links på internettet.

- [www.ctan.org](http://www.ctan.org) ( Comprehensive T<sub>E</sub>X Archive, når der nogen steder står CTAN:fuckinglangadresse, så mener de egentligt [www.ctan.org/fuckinglangadresse](http://www.ctan.org/fuckinglangadresse))
- The not so short guide to L<sup>A</sup>T<sub>E</sub>X, af Tobias Oetiker.  
CTAN:/tex-archive/info/lshort/english (klart anbefalet, faktisk et must)
- T<sub>E</sub>X unbound, L<sup>A</sup>T<sub>E</sub>X & T<sub>E</sub>X Strategies for Fonts, Graphics & more, af Alan Hoenig. ISBN:0 – 19 – 509685 – 1 (omhandler T<sub>E</sub>X og er ikke en begynderbog)
- [www.miktex.org](http://www.miktex.org) (Her kan man hente en compiler, gratis)
- <http://www.toolscenter.org/products/texniccenter/> (her kan man hente T<sub>E</sub>XnicCenter, gratis)