

Vildere L^AT_EX!

Tips, tricks og store projekter

Steffen Videbæk Petersen
latex@spet.dk

26. februar 2013

1 / 67

Del I

HverdagsL^AT_EX

2 / 67

1. halvdel: HverdagsL^AT_EX

1 Valg af editor

Hvad en god editor kan gøre for dig
Mine forslag til dig
Valg af PDF-fremviser

2 Hvad du måske allerede ved

L^AT_EX på den "rigtige" måde
Dokumentklasser og uundværlige pakker

3 Nye makroer og environments

Hvorfor bruge tid på det?
Egne makroer: Sådan
Egne environments: Sådan

3 / 67

Hvad betyder valget?

En rigtig god editor vil

- give færre dumme fejl, fx manglende }
- give færre anslag på tastaturet
- give færre referencefejl
- give bedre overblik over koden
- gøre det enklere at finde og rette fejl

Summa summarum

En god editor vil spare dig tid og frustrationer. Den vil gøre det nemmere og mere behageligt at skrive dokumenter i L^AT_EX.

4 / 67

Den gode editor *skal* kunne

- Syntaks-fremhævnning
- Syntax completion
- Liste over labels
- Understøtte dokumenter delt i flere filer
- Compile og trække fejlene ud af loggen

5 / 67

Derudover *bør* den kunne

- Outline dit dokument
- Understøtte PDF-synkronisering (SyncTeX)
- Tilbyde stavekontrol
- Have skabeloner for indsættelse af figurer
- Hjælpe med at lave tabeller
- Lave genveje til ofte brugte makroer og environments

6 / 67

Men fremfor alt

Du skal kunne bruge den effektivt!

Råd: Find den rette ditor

Brug den tid, det tager at finde den rigtige editor. Brug også tid på at lære den at kende og tid på at indstille den efter dine behov. Tiden du bruger her, sparer du tilfoldigt!

7 / 67

For alle: TeXstudio

TeXstudio fordele

- Kan det, den skal
- Kan det, den bør
- Overskueligt interface
- Enkel opsætning
- Virker på både Windows, Mac og Linux

Ulemper

- Begrænsede skabeloner
- Kender ikke fx beamer- eller memoir-makroer

8 / 67

For de modige: emacs eller vim

Emacs' og vims fordele

- Kan det hele – *alt* – også beamer- og memoir-makroer
- Uovertrufne udvidelsesmuligheder
- Uovertrufne konfigurationsmuligheder
- Kan bruges til meget andet end \LaTeX
- Virker på både Windows, Mac og Linux

Ulemper

- Stejl indlæringskurve
- Besværlig opsætning, specielt af Sync \TeX og stavekontrol

9 / 67

Glem ikke at vælge PDF-fremviser

Undgå Adobe Reader

- Langsom til at vise filerne
- Låser din PDF-fil
- Besværliggør opsætning af editor
- Kan ikke sættes op til Sync \TeX

Råd: Brug TeXstudios indbyggede eller Sumatra PDF

- Låser ikke din PDF-fil
- Overvåger din PDF-fil og opdaterer visningen

10 / 67

\LaTeX er ikke en brødrister, men...



XKCD 293: RTFM

Råd: Lær \LaTeX at kende

\LaTeX er et power tool. 10 minutter brugt på at læse dokumentation kan spare dig for timers arbejde og frustrationer senere hen.

Råd: Brug \TeX Lives texdoc

Du kan læse dokumentation ved at køre texdoc *<pakke>* fra Startmenuens søgefelt (eller ækvivalent).

11 / 67

Om dokumentation

- Ofte meget omfattende
- Nogle gange formanende
- Til tider uforståelig

Råd: Kend din dokumentation

Nogle manualer, fx den til memoir-klassen, er meget omfattende. Hvis du mangler en overspringshandling, så scroll igennem den. Det er en god måde at få overblik på.

12 / 67

Målet helliger midlet

Der er mere end én måde

Der findes meget ofte mere end én måde at gøre tingene på, og nogle vil være mere “rigtige” end andre. Du kan måske, i fremtiden, drage fordel af at gøre det “rigtigt” allerede nu, men er tiden til det?

Råd: Hacks er ok!

Hvis du er tilfreds med resultatet, så ignorer \TeX perter, der prædiker den “rigtige” løsning (også mig!).

Råd: ... oftest

Den “rigtige” løsning kan spare dig tid senere, så måske du skulle spørge \TeX perten igen, når du har tid?

13 / 67

Der er en nemmere måde

Hvis du tænker

- Det kan da ikke passe, at jeg skal gøre det *hver gang*?
- Hvorfor skal jeg copy/paste så meget kode?
- Det her kode har gigant-SARS! Aaaaargh!

Råd: Spørg en \TeX pert til råds

Identificer det, du gentager igen og igen eller hvad du synes er grim kode. På nettet eller på uni findes en \TeX pert, der har gjort det før, eller som er villig til at hjælpe dig med at finde den nemmere måde.

14 / 67

memoir til print

Råd: Brug memoir til print-dokumenter

Når først du har lært at bruge memoir er der ikke behov for at bruge andre dokumentklasser.

Hvad memoir kan

- Mange indbyggede pakker, ofte med udvidet funktionalitet
- Mange ting er enklere, fx sidehoveder og -fodder
- Ændring af virkemåde er enklere, fx nummerering
- Omfattende dokumentation
- Let at få yderligere hjælp

15 / 67

beamer til slides

Råd: Brug beamer til slides

Disse slides er lavet i beamer. Koden kan findes på nettet og viser, hvor simpelt det er.

Slides er et foredrag for sig

At lave slides i \LaTeX er ikke svært; med beamer-klassen er det ret enkelt og dokumentationen giver en grundig introduktion.

16 / 67

Pakken microtype

Vi giver en opsummering af arbejdet i det forløbne år og fortæller om vores planer for næste år, hvor vi selvfølgelig også modtager dine gode ideer. Har du fx ideer til, hvordan det sociale eller fysiske studiemiljø kan forbedres eller hvordan undervisningen skal udvikles, så har du nu muligheden for at få en finger med i spillet. Der bliver lyttet til MFSR, og vi har repræsentanter i mange råd og nævn, så hvis du har gode ideer, er der god mulighed for, igennem MFSR, at få gjort noget ved dem og få forbedret dit eget studie.

Uden microtype

Råd: Brug microtype

Pakken sørger for at mikrojustere afstanden mellem bogstaver og kan placere tegnsætning (delvist) i marginen. Det kan spare dig for orddeliger. Er særdeles anvendelig i flerspaltede dokumenter.

Med microtype

Vi giver en opsummering af arbejdet i det forløbne år og fortæller om vores planer for næste år, hvor vi selvfølgelig også modtager dine gode ideer. Har du fx ideer til, hvordan det sociale eller fysiske studiemiljø kan forbedres eller hvordan undervisningen skal udvikles, så har du nu muligheden for at få en finger med i spillet. Der bliver lyttet til MFSR, og vi har repræsentanter i mange råd og nævn, så hvis du har gode ideer, er der god mulighed for, igennem MFSR, at få gjort noget ved dem og få forbedret dit eget studie.

ams-pakkerne

Råd: Inkluder altid ams-pakkerne

Pakkerne amsmath, amssymb og amsfonts giver dig mulighed for at skrive matematiske udtryk – noget de færreste kan være foruden.

Dokumentation vs. daleif

Dokumentationen for ams-pakkerne er omfattende, men *ikke* brugervenlig. Kig istedet daleifs bog igennem for eksempler på crazy matematik. Hvis du vil gøre det, er det med garanti blevet gjort før.

Pakken graphicx

Råd: Brug pakken graphicx og læs dens dokumentation

Pakken tillader dig at indsætte billedfiler i dine dokumenter. Der er dog mange konfigurationsmuligheder, som de færreste kender til.

Global konfiguration (i preamble)

- Led efter billedfiler i specifikke mapper:
`\graphicspath{{figurer/}{spektre/}}`
- Sæt bredde for alle inkluderede billeder:
`\setkeys{Gin}{width=0.8\textwidth}`

Pakken mathtools

Bredt indeks...

$$V = \sum_{1 \leq i \leq j \leq n} v_{ij}$$

```
V = \sum_{1 \leq i \leq j \leq n} v_{ij}
```

med \mathclap

$$V = \sum_{1 \leq i \leq j \leq n} v_{ij}$$

```
V = \sum_{1 \leq i \leq j \leq n} \mathclap{v_{ij}}
```

Råd: Brug mathtools

mathtools tilføjer nogle muligheder, som ikke findes i ams-pakkerne. Mange af eksemplerne på avanceret matematik i daleifs bog bruger pakken. Manualen er ganske god og fyldt med eksempler.

Pakken siunitx

Råd: Brug siunitx

Bruger du tal eller enheder kan du drage fordel af siunitx, der gør det nemmere at være konsistent. Derudover tilføjer den nogle uundværlige værktøjer for datatabeller.

Eksempel

- $\$g=\text{SI}\{9.82\}\{\text{kilo}\backslash\text{gram}\backslash\text{per}\backslash\text{second}\backslash\text{squared}\}\$$
 $g = 9.82 \text{ kg s}^{-2}$
- $\text{SI}\{3,23\}\{\text{kg}\backslash\text{J}^{-1}\}\backslash\text{angstrom}\}$
 $3.23 \text{ kg J}^{-1} \text{ \AA}$
- $\text{SI}\{\text{per-mode}=\text{fraction}\}\{1\text{e-}34\}\{\text{joule}\backslash\text{per}\backslash\text{mole}\backslash\text{per}\backslash\text{kelvin}\}$
 $1 \times 10^{-34} \frac{\text{J}}{\text{mol K}}$
- $\text{num}[\text{separate-uncertainty}]\{1.34(5)\text{e}4\}$
 $(1.34 \pm 0.05) \times 10^4$

21 / 67

Målet med egne makroer

Målet med egne makroer og environments er

at spare tid Færre tastaturtryk giver et mindre tidsforbrug:
 $\backslash\text{pcc}$ er hurtigere end »photocatalyst coating«

at give mere overskuelig kode $\backslash\text{inv}$ er nemmere at læse (og taste) end $\wedge\{-1\}$.

at givet et konsistent udseende Egen notation i egne makroer giver mulighed for at ændre notation for hele dokumentet ved at ændre én linje i preamblen.

Råd: Brug egne makroer

Det koster indledningsvis noget tid, men den får du igen i form af færre tastaturtryk og mere overskuelig kode.

22 / 67

Hvad koster det mig

Du sparer tid, men først efter du har investeret den i

- at tænke over, hvor du kan spare tiden
- at lære værktøjerne at kende
- kodning og debugging
- dokumentation

Råd: Dokumenter din egen kode

Når du laver en ny makro, så dokumenter den. Hvis ikke det virker, som det skal, kan du eller en $\text{T}_{\text{E}}\text{X}$ pert se, hvad idéen var. Hvis andre får din kode, kan de se, hvordan det skal bruges.

23 / 67

Midlerne

Der er mange muligheder for at skrive egne makroer:

Simple $\backslash\text{newcommand}$ med eller uden argumenter

Betingelser Pakkerne ifthen og xifthen

Avancerede Pakkerne xparse og etoolbox

Fortvivl ej!

Syntaksen for disse makroer og pakker er ganske ligetil. Det kræver ikke lang tid og utallige forsøg at lave de makroer, som de fleste skal bruge.

24 / 67

Makroen \newcommand

Syntaks for \newcommand

Uden argumenter er \newcommand en meget enkel makro:

```
\newcommand\nymakro{⟨handling⟩}
```

Eksempel (\pcc)

```
\newcommand\pcc{photocatalyst coating\hspace}
```

Råd: Brug pakken xspace

Makroen \hspace fra pakken xspace indsætter automatisk et mellemrum, hvis det skal være der. Det gør makroen nemmere at bruge.

25 / 67

Overskuelighed og nye makroer

Eksempel

Læsbar kode

- `\newcommand{\isomorphic}{\cong}`
- `\newcommand{\DistAs}{\sim}`
- `\newcommand{\ObsFromDist}{\mathrel{\sim\sim}}`

Råd: Læsbarhed er vigtigere end antal anslag

Læsbar kode er enkel at debugge. Det er derfor en væsentlig fordel, at koden er læsbar. \isomorphic er længere end \cong, men øger læsbarheden og er derfor en god makro.

26 / 67

\newcommand med argumenter

Syntaks for \newcommand

Med argumenter er det ikke meget sværere:

```
\newcommand\nymakro[⟨antal args⟩]{⟨handling⟩}.
```

I handlingen refereres til argumenter med #n.

Eksempel (Partielt afledte)

Et ∂ laves med \partial, så partielt afledte er trælse at skrive, fx kræves $\frac{\partial}{\partial x}$. Med

```
\newcommand\pdiff[2]{\frac{\partial #1}{\partial #2}}
```

kan det nu laves med \pdiff{h}{x}.

27 / 67

Eksempler på konsistent notation

Konsistent notation, med mulighed for global ændring, er fx indre produkt, norm og absolutværdi:

Kode: Notationsskift med ét %-tegn

```
\newcommand{\inner}[2]{\langle #1, #2 \rangle}
% \newcommand{\inner}[2]{(#1; #2)}
\newcommand{\norm}[1]{\lVert #1 \rVert}
% \newcommand{\norm}[1]{\lvert #1 \rvert}
\newcommand{\abs}[1]{\lvert #1 \rvert}
```

Råd: Brug dine makroer konsekvent

Hele idéen bag at bruge makroer til notation forsvinder, hvis ikke man bruger dem hver gang. Hvis du laver en makro for fx indre produkt skal du også bruge den hver gang du opskriver et indre produkt.

28 / 67

\newcommand med valgfrie argumenter

Syntaks

```
\newcommand\nymakro[⟨args⟩][⟨default⟩]{⟨handling⟩}
```

Der kan kun angives ét valgfrit argument, og det er det første.

Eksempel (Valgfrie argumenter)

Ved at ændre definitionen af \pdiff til

```
\newcommand\pdiff[3][\partial]{\frac{#1 #2}{#1 #3}}
```

\pdiff{h}{x} giver stadig $\frac{\partial h}{\partial x}$ men \pdiff[d]{h}{x} giver $\frac{dh}{dx}$.

Råd: Valgfrie argumenter giver fleksibilitet

Man kan opnå større fleksibilitet med valgfrie argumenter. Pakkerne xparse og etoolbox giver modsat \newcommand mulighed for mere end ét valgfrit argument.

29 / 67

Makroen \newenvironment

Syntaks

Meget tæt opad \newcommand, men tager en start- og en sluthandling:

```
\newenvironment{⟨env⟩}{⟨start⟩}{⟨slut⟩}
```

```
\newenvironment{⟨env⟩}[⟨args⟩]{⟨start⟩}{⟨slut⟩}
```

```
\newenvironment{⟨env⟩}[⟨args⟩][⟨default⟩]{⟨start⟩}{⟨slut⟩}
```

Pas på!

Du kan ikke brug argumenterne i sluthandlingen. Løsningen er at bruge pakken xparse hvis det er nødvendigt.

30 / 67

Eget citations-environment

Kode: Eget environment til citationer

```
% Environmentet myquote, der tager et argument: Hvem der
% citeres. Dette sættes i fed paa en centeret linje over
% selve citatet, der indrykkes fra begge marginer og
% goeres slanted (dvs. skraa, men ikke kursiv).
\newenvironment{myquote}[1]%
{\centerline{\bfseries Citat fra #1}\begin{quote}\slshape}%
{\end{quote}}
```

Råd: Lange linjer vs. %-tegn

Din preamble kan let blive uoverskuelig hvis linjerne bliver alt for lange. I nogle tilfælde bliver L^AT_EX dog arrig, hvis den møder et uventet linjeskift i koden. Brug % til at skjule linjeskiftet.

31 / 67

Eget citations-environment II

Kode: Navn nederst med hjælp fra xparse

```
% Ny version af myquote, hvor argumentet til myquote
% skal bruges i sluthandlingen.
\NewDocumentEnvironment{m}%
{\begin{quote}\slshape}%
{\nopagebreak\vskip\baselineskip\hfill--#1}
```

Råd: Spacing og environments

I og omkring environments opfører afstande sig sært; det kan være svært at regne ud, hvad der sker og hvorfor. Lad være med at tænke over det i starten. Hvis du synes det ser grimt ud, så snak med en T_EXpert om det.

32 / 67

(Næsten) kun fantasien sætter grænser

Med list, snedighed og brug af forskellige pakker kan man næsten alt:

- Lave rammer omkring stort set alt, også andet end bare lige linjer, og måske kun ramme i top og bund?
- Ændre baggrunds- og tekstfarver
- Behandle samme argument som både tekst og makro
- Automatisere fx tabelgeneration
- ...

Råd: Drøm og spørg til råds

Der er intet i vejen for at ønske noget bestemt og i de fleste tilfælde kan det også lade sig gøre. T_EXperterne hjælper gerne men prædiker måske også lidt (undskyld).

33 / 67

Del II

Store projekter

34 / 67

2. halvdel: Store projekter

4 Dokumenter i flere filer

Opdelingen
Inklusion af filer
Selektiv inklusion

5 BibT_EX

Hvad det er
Inklusion i dokument

6 Mit forslag til en skriveproces

Form vs. indhold
Gode værktøjer
Undervejs
Til sidst

35 / 67

En fil, to filer, ..., n filer

L_AT_EX kan inkludere indholdet af andre filer. Det kan bruges til at skabe

- mere overskuelighed
- enklere debugging
- kortere compile time

Nogle pakker tillader, at man henter input fra andre filer, fx listings.

36 / 67

Hvornår skal man opdele?

Hvornår skal man dele noget mellem flere filer?

- Preamblen i sin egen fil
- Hvert kapitel i sin fil
- Uoverskuelige tabeller/figurer i egne filer

Råd: Mådehold i opsplitning

Man kan laver filer for hvert afsnit, eller hver subsection eller hver tabel etc. På et tidspunkt bliver det mindre overskueligt, så overvej hvor du deler.

37 / 67

Opdeling og TeXstudio

Master-fil

Mange editorer bruger udtrykket master-fil. Det er den fil, der kalder `\include` for de andre filer. Det er en god idé, at din preamble er din masterfil og alt indhold er i andre filer.

Et eksempel

38 / 67

Makroen `\include`

Makroen `\include` indsætter indholdet af en fil men starter også en **ny side**.

Syntaks

`\include{⟨filnavn uden .tex⟩}`
Filens navn *skal* slutte på `.tex`.

Råd: `\include` hvornår?

Brug `include` når du skal starte på en ny side. Skriver du i memoir er det for hvert `\chapter`.

39 / 67

Makroen `\input`

`\input` gør som `\include` men starter *ikke* en ny side.

Syntaks

`\input{⟨filnavn uden .tex⟩}`
Filens navn *skal* slutte på `.tex`.

Råd: `\input` hvornår?

Brug `\input` til fx tabeller, figurer eller andet, der skal skjules i overskuelighedens navn.

40 / 67

Inklusion af kode med listings

Kode: \input af linjerne 7–18

```
\lstinputlisting[firstline=7,lastline=18]{isPrime.cpp}
```

Eksempel

```
bool isPrime (int n) {  
    // A primitive trial division  
    if (n <= 1)  
        return false;  
    if (n % 2 == 0)  
        return (n == 2);  
    for (int d = 3; d <= sqrt(n); d++) {  
        if (n % d == 0)  
            return false;  
    }  
    return true;  
}
```

41 / 67

Indsættelse af data i tabeller

Man kan bruge \input til at slippe for at copy/paste. Hvis man fx vil inkludere data fra filen data.dat kan man:

Kode: \input vs. copy/paste

```
\begin{tabular}{r r r}  
    \toprule  
    Vinkel & Kanal 1 & Kanal 2 \\  
    \midrule  
    \input{data.dat}  
    \bottomrule  
\end{tabular}
```

Dataformat

\input{data.dat} virker kun, hvis dataene i data.dat er i L^AT_EXs tabelformat.

42 / 67

Inklusion af fremmede formater

Råd: Brug pakken datatool

Pakken datatool kan hente data fra eksterne filer og formater dem anderledes. Den kan også hente specifikke søjler eller rækker.

Råd: Kombiner datatool og siunitx

Hvis man kombinerer datatool med siunitx kan man lave vilde ting. Fx kan man hente data fra en ekstern fil, der ikke er i L^AT_EX-format og endda omskrive $1.03e3$ til 1.03×10^3 .

43 / 67

Spas noget compile-tid væk

Makroen \includeonly benyttes i stedet for at kommentere de enkelte \include-statements ud. Den

- Inkluderer kun de nævnte filer
- Benyttes i preamblen
- Beholder referencerne til de filer, der ikke compiles

Kode: Kun eksperiment.tex compiles

```
\includeonly{eksperiment} % Kommasepareret liste af filer  
  
\begin{document}  
    \include{intro}  
    \include{eksperiment}  
    \include{resultater}
```

44 / 67

Effektiv brug af \includeonly

Råd: Brug af \includeonly

Brug \includeonly til at minimere den tid, du bruger på at compile. Ved kun at inkludere de filer, du laver ændringer i, bliver processen lidt mere flydende.

Idet \includeonly husker labels fra de ikke-inkluderede filer minimerer man også referencefejlene på tværs af kapitler.

\includeonly og \input

\includeonly er smart, men virker kun for filer inkluderet med \include. Filer inkluderet med \input indsættes altid.

45 / 67

Hvad er BibTeX

BibTeX er

- et program, der lever uafhængigt af L^AT_EX
- en database over det litteratur, du har læst

BibTeX kan

- give en konsistent visning af referencer til det, du citerer
- styles, så den viser fx [1] eller [Jackson, 1998]
- opsættes til at vise alle værker eller kun dem, du citerer

46 / 67

Databasens indhold

- Man kan ikke generere en bibliografi uden at have noget at citere. Man skal derfor opbygge en fil med værkerne. Filen skal have endelsen .bib, fx spet.bib
- Man kan skrive informationerne i hånden, men man kan også finde informationerne på nettet. De fleste naturvidenskabelige artikeldatabaser giver mulighed for at downloade en .bib-fil tilhørende hver artikel
- Din BibTeX fil kan indeholder langt mere information om hvert værk, end der skrives i litteraturlisten. Det er stilarten, der bestemmer hvilke informationer, der medtages.

Råd: Byg din .bib-fil fra dag ét

At skulle lave citeringer som noget af det sidste er besværligt. Hvor var det lige man læste hvad henne? Derfor er det en god idé at opbygge sin .bib-fil fra starten og citere løbende.

47 / 67

Eksempel på en .bib-fil

Kode: Uddrag fra Rasmus Villemoes' thesis.bib

```
@book {Hatcher2002,
  AUTHOR = {Hatcher, Allen},
  TITLE = {Algebraic topology},
  PUBLISHER = {Cambridge University Press},
  YEAR = {2002},
  PAGES = {xii+544},
  ISBN = {0-521-79160-X; 0-521-79540-0},
  MRCLASS = {55-01 (55-00)},
  MRNUMBER = {MR1867354 (2002k:55001)},
  MRREVIEWER = {Donald W. Kahn},
}
@article {MR2339577,
  AUTHOR = {Andersen, J{\o}rgen Ellegaard and Ueno, Kenji},
  TITLE = {Abelian conformal field theory and determinant bundles},
  JOURNAL = {Internat. J. Math.},
  FJOURNAL = {International Journal of Mathematics},
  VOLUME = {18},
  YEAR = {2007},
  NUMBER = {8},
  PAGES = {919--993},
  ISSN = {0129-167X},
  MRCLASS = {81T40 (57R56 81T45)},
  MRNUMBER = {MR2339577},
}
```

Her er Hatcher2002 og MR2339577 citeringsnøgler.

48 / 67

Hvordan bruges BibTeX

- Citationer laves i teksten med `\cite{⟨nogle⟩}`, fx
`\cite{MR2339577}`
`\cite{MR2339577,Hatcher2006}`
- I ens dokument skriver man
`\bibliographystyle{alpha}`
`\bibliography{thesis}`
der hvor bibliografien skal indsættes.
- For at generere bibliografien skal man køre kommandoen
`bibtex master`
- Næste gang man compiler `master.tex` indsættes bibliografien for de elementer, der er blevet citeret.

49 / 67

Løst og fast om BibTeX

Råd: Ikke-citerede værker

BibTeX inkluderer kun det, der nævnes med `\cite`. Med `\nocite{⟨nogle⟩}` inkluderes også dét værk uden citation. Med `\nocite{⟨*⟩}` inkluderes alt fra `.bib`-filen.

Råd: Bibliografistil

En bibliografistil styrer hvordan både citationer og litteraturliste kommer til at se ud og der er mange at vælge imellem. Pakken `natbib` er et godt sted at starte.

Råd: Citationer som en del af teksten

`\cite{⟨nogle⟩}` giver fx `[3]` eller `[Jackson, 1998]` men hvis citationen er den del af teksten bruges `\citet`. Den giver `Jackson (1998)`, der kan læses som en del af teksten.

50 / 67

Skriv og leg hver for sig

Råd: Skriv!

Der skal nok komme tidspunkter, hvor du sidder med en skriveblokada. Skriv, når du kan. Så kan du bruge blokeringsperioder til at nørde L^AT_EX og lege med layoutet.

Råd: Leg!

Når du sidder med en skriveblokada, så kan du (prøve at) lege med layoutet. Kig i forskellige lærebøger og søg inspiration. Det meste kan lade sig gøre og det er en god måde at lære på.

Råd: Udkommenter det, der ikke virker

Ved at udkommentere kan du bruge tid på at skrive fremfor på at løse L^AT_EXniske problemer. Problemerne kan du altid vende tilbage til og du kan altid spørge en T_EXpert til råds senere.

51 / 67

Skabelon eller ej?

Der er fordele ved at bruge en god skabelon

- Den sparer dig for meget tid i forhold til at sætte en preamble op fra bunden
- Man kan skele til de løsninger, der er brugt i den, og bruge dem som de er eller som inspiration

Men der er også ulemper

- En udokumenteret skabelon kan spilde din tid
- En inkomplet eller uddateret skabelon kan være en hæmsko for hele skriveprocessen
- Det kan være svært at se, hvad der er en god skabelon

52 / 67

Når du vælger en skabelon

Råd: Dokumentation

Kig efter dokumentation, særligt i preamblen. Ingen dokumentation er som regel et tegn på, at den kan blive tung at danse med.

Råd: Overskuelighed

Hvis preamblen virker fuldstændig uoverskuelig er du nok også på vej ud i noget snavs. Hvis ikke der er dokumentation må du snakke med forfatteren og høre, hvad han/hun har lavet.

Råd: Minimalisér skabelonen

Din første opgave bør være at pille alt, du ikke skal bruge, ud af skabelonen. Når du har gjort det, skal du have den til at virke. Lav et eksempel der kan compile.

53 / 67

Den gode skabelon

Den gode skabelon har nogle helt bestemte egenskaber

- Den er skrevet i memoir
- Den er veldokumenteret, både i preamble og dokument
- Preamblen er velstruktureret
- Der er kontaktoplysninger til forfatteren, så man kan få hjælp hvis kommentarerne ikke er nok

Råd: Brug Rasmus Villemoes' skabelon

Rasmus Villemoes er T_EXpert. Han brugte sin ph.d.-afhandling på at skrive en god og veldokumenteret skabelon. Den kan være overkill, men er som minimum et godt sted at hente inspiration.

<http://rasmusvillemoes.dk>, og download L^AT_EX-koden fra hans afhandling.

54 / 67

Pakken fixme

fixme giver mulighed for

- at lave noter i margin
- at lave en liste over noter
- at L^AT_EX giver en compile-fejl, hvis der er noter tilbage

Råd: Brug fixme

Brug pakken fixme til at holde styr på de ting, der skal undersøges nærmere eller gøres noget ved senere. Det giver dig mulighed for at skrive nu og tage dig af fejlene senere.

55 / 67

Pakken showkeys

showkeys giver mulighed for

- at vise labels i din PDF-fil
- at få overblik over, hvor man refererer forkert
- at holde styr på labels, hvis ens editor ikke gør det

Råd: Brug showkeys

Brug showkeys undervejs i skrive- og retteprocessen til at holde styr på labels og referencer. Den er et uvurderligt værktøj når man skal fejlfinde referencer.

56 / 67

Pakke lipsum

Hvis man eksperimenterer med layout er pakken lipsum fantastisk. `\lipsum` indsætter ca. halvanden sides fyldtekst og med valgfrit argument indsætter den et eller flere afsnit, fx `[1-3]` eller `[1,5]`:

Eksempel (`\lipsum[4]`)

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

57 / 67

Snedigere referencer med pakken varioref

Pakken varioref giver mulighed for snedigere referencer. Hvor `\pageref{<label>}` skriver et sidetal for denne label, vil `\vpageref{<label>}` skrive fx:

- På foregående side
- På næste side
- Herunder

Råd: Brug varioref

Hvis du har mange referencer til særligt figurer og tabeller er det en fordel at bruge varioref, specielt hvis de er lokale. Dokumentet bliver meget nemmere at læse.

58 / 67

Kodehygiejne

- Selv hvis du er den eneste, der skal læse din kode, så skal du kunne finde rundt
- `\begin` og `\end` på linjer for sig
- Mellumrum i matematik ignoreres, hvilket kan gøre koden mere læsbar. Fx:

```
$(a_1, b_1 + n a_1, a_2, b_2, \ldots, a_g, b_g)$
$(a_1, b_1 + n a_1, a_2, b_2, \ldots, a_g, b_g)$
```

- Antallet af tomme linjer mellem afsnit er irrelevant, så lav gerne et par tomme linjer inden `\section` og lignende.
- Dokumentér særligt preamblen og hacks.

59 / 67

Labels defineret to gange

Dette er en klassisk fejl og god grund til at bruge `\includeonly`

```
LaTeX Warning: Label 'eq:einstein' multiply defined.
...
LaTeX Warning: There were multiply-defined labels.
```

Råd: Ret dobbeltlabels med det samme

Ret den med det samme, ellers kan du ikke huske, hvilken ligning du egentlig ville referere til. For hver `\eqref{eq:einstein}` skal du finde ud af hvilken du egentlig mente.

60 / 67

Labels, der ikke er definerede

En ligeså klassisk fejl er:

```
LaTeX Warning: Reference 'eq:32' on page 1 undefined on input line 11.  
...  
LaTeX Warning: There were undefined references.
```

Råd: Ret ikke-definerede labels med det samme

Hvis fejlen ikke forsvinder ved næste compile, så har du henvist til en ligning der ikke findes. Ret mens du kan huske hvad du ville referere til.

61 / 67

Sidens layout

Indhold, fonte og marginer er ok. Næste skridt er layoutet:

- Kapitelforsider
- Environments, fx sætningskonstruktioner
- Udseendet af hjemmelavede makroer
- Tabeller, figurer og andre flydende elementer

Råd: Layout og fremtoning

Sidens layout betyder rigtigt meget for hele projektets fremtoning. Hvad der er rigtigt for forfatteren af din skabelon er ikke nødvendigvis rigtigt for dig. Tag en snak med en TeXpert, gerne et stykke tid inden deadline.

63 / 67

Fonte og marginer

Når indholdet er færdigt skal følgende afgøres:

Skrifttype L^AT_EXs standardfont, Computer Modern, er ikke eneste valg. Se fx www.tug.dk/FontCatalogue.

Skriftstørrelse Typisk 10–12pt, men andre fonte kan se bedre ud i andre størrelser.

Papirstørrelse For de fleste A4, men der kan være krav alt efter hvad du skriver, særligt ph.d.-afhandlinger findes i B5.

Marginer Brede marginer giver korte linjer, og det er med vilje. Lange linjer er sværere at læse end korte.

62 / 67

Fejl og advarsler

Indhold og al layout er på plads. Nu skal det sikres at:

- Dokumentet skal kunne oversættes uden fejl. daleif har en liste over almindeligt forekommende fejlmeddelelser og hvad de betyder på dansk.
- Rent faktisk bør fejl rettes så snart de opstår; så har man lettere ved at finde den skyldige kodebid. Jo mere kode man fylder på, jo sværere er det at afkode L^AT_EXs til tider kryptiske meddelelser.
- Advarsler skal man tage alvorligt. Nogle kan ignoreres, men det bør være et aktivt valg for hver enkelt.

64 / 67

Almindelige advarsler

De mest almindelige advarsler er (under/over)full (h/v)box:

Underfull hbox

L^AT_EX strækker en linje mere end den bryder sig om. Er der et ord, den ikke kan dele eller har du måske brugt \\ til at dele en linje med? (Fy, skam dig!)

Overfull hbox

Den mest almindelige advarsel af alle. L^AT_EX brokker sig over, at noget tekst stikker ud i marginen. Du hjælpe den med at foretage orddeling, omskrive teksten eller acceptere overskridelsen.

65 / 67

Fejl, advarsler og badboxes

Råd: Sæt tid af til at rette fejl og advarsler

Specielt for store dokumenter kan det være lidt af en opgave, at rette fejl og advarsler. Hvis man bruger en ringe skabelon vil det også kræve ekstra tid. Sørg for at have sat tid af inden din deadline.

Råd: Badboxes er ikke ligegyldige

Badboxes vidner om, at noget er galt. Det er ikke altid man helt kan lure, hvad det er L^AT_EX brokker sig over, men det skal være et valg at ignorere den enkelte badbox

66 / 67

Mit bedste og sidste råd

Råd: Kom i gang

Jo længere noget udskydes, desto sværere bliver det at komme i gang med. Spar dig selv frustrationer:

- Skriv, selvom det ikke kan compile.
- Skriv, selv hvis ikke du har en preamble.
- Brug dine egne makroer, selvom de ikke helt virker.
- Citer, selvom du ikke kender BibTeX.
- Spørg om hjælp tidligt.

67 / 67