

LaTeX für Lehrer

Sachverstand2 @ Geocities.com

12. Februar 2006

Inhaltsverzeichnis

1	Intro	2
2	Paradigma	3
3	Workstation	4
4	Das erste LaTeX-Dokument	5
5	Dokumente erschaffen	6
5.1	tex	6
5.2	latex	6
5.3	pdflatex	6
5.4	Es klappt nicht beim ersten Mal	6
6	Organisieren des Inputs	7
7	Reservierte Zeichen und ihre Bedeutung	9
8	Tabellen	10
8.1	In Prosa	10
8.2	In LaTeX	10
8.3	In real	11
9	Listen	12
10	Bilder	13
10.1	Konvertieren von gif und jpeg nach.png	13
10.1.1	Auf der Kommandozeile	13
10.1.2	Mit ImageMagick	13
10.2	Einbinden	13
10.3	Und so sieht's aus	13
11	Arbeitsblätter einbinden	14
12	Happy LaTeXing!	15

<i>INHALTSVERZEICHNIS</i>	2
13 Anhang 1 : Kommentiertes Listing	16
14 Anhang 2 : Skripten	19
14.1 Verzeichnisstruktur und Prüfung	19
14.2 Ein paar Skripten	21
14.2.1 Skripting	21
14.2.2 Bilder einbinden	21
14.2.3 Die große Sause	22
14.2.4 .fig braucht eine spezielle Behandlung	22

1 Intro

Standard-Textprozessoren haben große Nachteile:

- sie fressen viel Speicherplatz;
- sie machen vieles automatisch falsch;
- sie zwingen zum Aufbewahren alter Versionen (falls die soeben gespeicherte Datei, zB durch einen Absturz, unbrauchbar wird);
- die Platzierung von Bildern erfolgt "auf der Seite festgenagelt" oder so unglücklich, daß viele halbleere Seiten entstehen;
- die Dokumente sehen auf jedem Drucker anders aus.

Gelegentlich beobachten wir auch Vorteile:

- alles steckt in einer Datei statt in vielen;
- man kann Bilder ohne Nachzudenken importieren.

Den ersten Vorteil erschlagen wir, indem wir pdf-Dateien erzeugen.

Ums Nachdenken bei den Bildern kommt man nicht so einfach herum, da muß man das entsprechende Kapitel dieses Tutorials lesen.

2 Paradigma

Textprozessoren verleiten den Benutzer, die zahlreichen vorhandenen Layout-Elemente auch zu benutzen. Die entstehenden Dateien werden deshalb schnell unübersichtlich.

LaTeX benutzt ein professionelles Paradigma:

- Ein Dokument besteht aus Sektionen, Subsektionen, SubSubsektionen, Paragraphen, Autor, Titel, Zusammenfassung, Titelseite, Inhaltsverzeichnis, Bildern, Tabellen, fertig. (Bibliografie lasse ich weg.).
- Bilder und Tabellen sind floats. Floats haben eine Bild- bzw. Tabellenunterschrift und sind numeriert. Wenn das float noch auf diese Seite paßt, dann kommt es dorthin, sonst kommt es eben auf die nächste Seite.
- Änderungen des Layouts erzwingen eine Reformatierung des gesamten Dokuments, incl. ggf. Neu-Anordnung der Bilder und Tabellen.

Der geneigte Leser wird an dieser Stelle bemerken, daß Latex sich eher für längere Texte eignet - zB Unterrichtsvorbereitungen für ein komplettes Halbjahr, oder Bücher.

Arbeitsblätter unter LaTeX zu erstellen ist dagegen eine Strafe. Es gibt allerdings Mittel und Wege, sie (selbst unter windos) zu erstellen und anschließend unter Benutzung von Open-Source-Software nahtlos ins pdf-Dokument einzubinden.

3 Workstation

Zum Erstellen professioneller pdf-Dokumente braucht man:

- IBM-PC oder kompatible, Pentium 100 oder mehr, 65MB RAM oder mehr, 1 GB Platte oder mehr;
- Monitor, Tastatur, Maus (für Preview);
- Redhat 6.2 Linux CD;
- wer drucken will: einen Drucker.

Vorbereitung: Von der CD booten, "Workstation installieren" anklicken, User "nemo" anlegen, Passwort nicht vergessen.

X einrichten: 800x600x256 Farben für 14"-Monitor, 1024x768x256 für 17"-Monitor.

Reboot, CD rausnehmen, fertig.

4 Das erste LaTeX-Dokument

Eine kommentierte Version hiervon und ein Screenshot findet sich im Anhang.

```

\documentclass [11pt] {article}

\usepackage {german}
\usepackage {umlaut}
\usepackage {palatino}
\usepackage {graphicx}
\usepackage {boxedminipage}
\usepackage [margin=14mm] {geometry}

\author {http://www.freewebs.com/gtslehrer}
\date {\today}
\title {Latex f"ur Lehrer - Test 1}

\pagestyle {headings}
\parindent 0.mm
\parskip=1mm
\fbboxsep=5mm
\fbboxrule=0.5mm

\newenvironment{lehr}{
  \begin{center}      \begin{boxedminipage}{\textwidth}
  \begin{bf}          \begin{large}
}{
  \end{large}        \end{bf}
  \end{boxedminipage} \end{center}
}

\begin{document}
\maketitle
\tableofcontents

\section{Einleitung}
bla bla bla bla bla bla bla bla bla bla bla bla bla

\section{Durchf"uhrung}
bla bla bla bla bla bla bla bla Ich fahre keinen Citro\"en.

\begin{lehr}
  Referentielle Integrit"at ist...
\end{lehr}

\section{Zusammenfassung}
bla bla bla bla bla bla bla bla bla bla bla bla bla

\end{document}

```

5 Dokumente erschaffen

Nach so viel Theorie wollen Sie sicher nun Ihre eigenen Dokumente erstellen.

Dazu brauchen Sie Ihren Satz Dateien (oder eine Datei mit allem drin - für den Anfang reicht das) und eine Kommandozeile.

5.1 tex

TeX von Donald Knuth ist der *godfather* des Dokumente-Erschaffens, aber es versteht kein LaTeX.

LaTeX ist allerdings ein Frontend zu TeX, und ist in TeX geschrieben.¹

5.2 latex

latex ist ein Programm, das eine Eingabedatei verlangt. Schreiben Sie also ein ganz kurzes `test1.tex`, und "compilieren" Sie es:

```
latex test1           # dabei kommt u.a. test.dvi heraus
dvips -f test1.dvi > test1.ps  # konvertiert nach postscript
ps2pdf test1.ps      # erzeugt test1.pdf
startx               # startet X
<klick>xterm         # startet ein xterm
xpdf test1.pdf       # startet den pdf-viewer.
```

5.3 pdflatex

pdflatex ist wiederum ein Frontend für LaTeX.

```
pdflatex test1      # dabei kommt u.a. test.pdf heraus
startx              # startet X
<klick>xterm        # startet ein xterm
xpdf test1.pdf      # startet den pdf-viewer.
```

5.4 Es klappt nicht beim ersten Mal

... und wenn Sie es trotzdem, nach Stunden, schaffen, haben Sie jetzt etwas Schönes verdient. ZB einen Kaffee oder einen Spaziergang oder ... na Sie wissen schon was Ihnen gut tut.

¹Puristen werden hier zürnen: TeX sei keine Programmiersprache.

Nun, TeX beschreibt, wie das Ergebnis aussehen soll. Programmiersprachen beschreiben, was das Produkt (das Programm) machen soll.

Da ein Dokument in erster Linie gut aussehen soll, ist TeX in diesem Sinne eine Programmiersprache. Das werden Sie spätestens an den ganzen Fehlermeldungen merken, die es während des Laufens ausspuckt, genau wie ein richtiger Compiler.

Korrekt beschrieben, ist TeX und sein frontend LaTeX natürlich eine Dokument-Beschreibungs-Sprache.

6 Organisieren des Inputs

Wenn Sie mal Ihre Lieblings-Seitenränder herausgefunden haben, wollen Sie sie automatisch in allen Dokumenten einfügen.

Textdateien werden aber schnell unübersichtlich, deshalb rate ich Ihnen:

- Verwenden Sie für jedes Dokument ein eigenes Verzeichnis.
Für dieses Dokument verwende ich gerade `/home/nemo/tex/141`.
- Verwenden Sie für jede Sektion eine Datei.
- Halten Sie Ihre Header (also die Datei, die alle dokumentenweiten Vereinbarungen enthält) in `/home/nemo/tex`.

Legen Sie im aktuellen Verzeichnis einen Softlink an, zB

```
cd /home/nemo/tex/141
ln -s ../header1.tex
ln -s ../header2.tex
```

Dieses Dokument sieht bisher so aus:

```
% this is 141.tex
\include{header1}

\title {LaTeX f"ur Lehrer}

\include{header2}

\include{intro}
\include{paradigma}
\include{workstation}
\include{1stlatex}
\include{orga}

\include{anhang}
\end{document}

% this is header1.tex
\documentclass [11pt, a4paper] {article}

\usepackage {german,umlaut}
\usepackage {palatino}
%\renewcommand{\rmdefault}{phv}
\usepackage {graphicx}
\usepackage {boxedminipage}
\usepackage [margin=14mm] {geometry}

\pagestyle {headings}
\parindent 0.mm
\parskip=1mm
\fbboxsep=5mm
```

```
\fboxrule=0.5mm

\newenvironment{lehr}{
  \begin{center}          \begin{boxedminipage}{\textwidth}
  \begin{bf}              \begin{large}
}{
  \end{large}            \end{bf}
  \end{boxedminipage}    \end{center}
}
```

```
% this is header2.tex
\author {http://www.freewebs.com/gtslehrer}
\date   {\today}

\begin{document}
\maketitle

\tableofcontents
```

Die Dateien werden durch `include` eingebunden.

Sie können auch `input` nehmen, aber dann müssen Sie das `clearpage` selbst hinschreiben.

Sie haben selbstverständlich schon gemerkt, daß alle Befehls Worte in LaTeX mit einem `\` (Backslash) anfangen, und Kommentare mit einem Prozentzeichen.

7 Reservierte Zeichen und ihre Bedeutung

LaTeX hat folgende reservierte Zeichen:

\$ % & ~ ^ _ \ { } \$ < >

- hash (Doppelkreuz): weiß ich nicht.

Dies ist ein Tutorial. Sie brauchen zum Arbeiten mit LaTeX eh ein Manual. Versuchen Sie es mit den exzellenten Seiten von Sheldon Green, die sind mit auf der RedHat 6.2-CD.

- Dollar: definiert Anfang und Ende der Mathematik-Umgebung. In einer Mathematik-Umgebung setzt TeX mathematische Formeln, und zwar perfekt.

Ist zu schwierig für ein Tutorial wie dieses.

- kaufm. Und (Ampersand) trennt Spalten in Tabellen.
- Tilde und Dach produzieren Buchstaben mit Tilde und Dâch.
- Dach und Grundstrich produzieren in Mathe-Umgebung Exponenten und Indizes.
- Backslash ist das Zeichen für "Hallo Latex, jetzt kommt ein Latex-Kommando." Doppel-Backslash bedeutet: Zeilenumbruch.
- Mengenklammern definieren Umgebungen und Kommandos.

Eine Umgebung beginnt mit `\begin{lehr}` Inneres der Umgebung `\end{lehr}`. Schauen Sie sich die Umgebung `lehr` und ihre Auswirkung im Anhang an.

Ein Kommando erstreckt sich bis zum Ende des Absatzes, bis zum Ende des Dokumentes oder bis zur nächsten rechten Mengenklammer. `{\em Hervorgehobener Text}` kommt so heraus: *Hervorgehobener Text*.

- größer und kleiner sind nur in der Mathe-Umgebung zulässig.

Aus `3 > $ 4` kommt heraus: $3 > 4$.

Falls Ihnen das nicht gefällt, versuchen Sie:

`$$ 3_{\text{langer Index}} > \frac{\text{zaehler}=4^{\{44\}}{\text{nenner}=\sqrt{2}} $$`

mit dem folgenden Ergebnis:

$$3_{\text{langerIndex}} > \frac{\text{zaehler} = 4^{44}}{\text{nenner} = \sqrt{2}}$$

oder

`$$ y_2 = \sqrt{\frac{1 + \frac{2}{3}}{1 - \sqrt{2}}} $$`

mit

$$y_2 = \sqrt{\frac{1 + \frac{2}{3}}{1 - \sqrt{2}}}$$

8 Tabellen

Eine Tabelle ist ein Float, d.h. sie "fließt" auf die nächste Seite, wenn sie auf die aktuelle nicht mehr paßt.

8.1 In Prosa

latex, hier kommt eine tabelle.

```
[
genau hierher,
oder wenn das nicht geht, unten auf der seite,
oder wenn das nicht geht, oben auf der seite,
oder wenn das nicht geht, auf einer extra-seite
]
die tabelle sieht so aus: {
ein senkrechter Balken,
eine zentrierte Spalte, so breit wie n"otig,
ein senkrechter Balken,
eine Spalte von halber \textwidth
}{
zun"achst eine horizontale linie
erste zelle
&
zweite zelle, da darf ich soviel text reinmachen wie ich will, LaTeX
schafft das schon, notfalls bricht es den Text um.

Ich darf sogar

mehrere Abs"atze da hineinstecken und nat"urlich

$$ e^{Formeln} $$

\hline \\

dritte zelle
&
vierte zelle

\hline \\

}
meine tabelle soll test-tabelle hei"sen.
ende der tabelle
```

8.2 In LaTeX

```
\begin{table} [hbt] % h here, b bottom, t top, p page
\begin{tabular}{|c|p{0.5\textwidth}}
\hline
erste zelle
&
```

zweite zelle, da darf ich soviel text reinmachen wie ich will, LaTeX schafft das schon, notfalls bricht es den Text um.

Ich darf sogar

mehrere Absätze da hineinstecken und natürlich

e^{Formeln}

`\hline`

dritte zelle

&

vierte zelle

`\hline`

`\end{tabular}`

`\caption{test-tabelle}`

`\end{table}`

8.3 In real

Siehe test-tabelle.

erste zelle	zweite zelle, da darf ich soviel text reinmachen wie ich will, LaTeX schafft das schon, notfalls bricht es den Text um. Ich darf sogar mehrere Absätze da hineinstecken und natürlich e^{Formeln}
dritte zelle	vierte zelle

Tabelle 1: test-tabelle

Der rechte senkrechte Strich fehlt, weil ich ihn im `tabular` nicht bestellt habe.

Das "Siehe test-tabelle" sieht merkwürdig aus. In LaTeX weiß ich aber nicht, ob die Tabelle auf derselben Seite landen wird.

Also darf ich nie schreiben: "Die folgende Tabelle zeigt...", sondern muß immer schreiben: "Tabelle (caption) zeigt..."

9 Listen

Selbsterklärend.

```

\begin{itemize}                                     % eine aufz"ahlung
  \item erster item
  \item zweiter item

  Neue Abs"atze werden korrekt einger"uckt.
  \item noch ein item
  \begin{enumerate}                               % eine Numerierung, geschachtelt
    \item erster numerierungseintrag % in die aufz"ahlung
    \item zweiter eintrag
    \item dritter

    auch dieser Absatz wird korrekt einger"uckt.
  \end{enumerate}
  \item der letzte eintrag der aufz"ahlung.
\end{itemize}

```

-
- erster item
 - zweiter item
 - Neue Absätze werden korrekt eingerückt.
 - noch ein item
 1. erster numerierungseintrag
 2. zweiter eintrag
 3. dritter
 - auch dieser Absatz wird korrekt eingerückt.
 - der letzte eintrag der aufzählung.

10 Bilder

Ich nehme im folgenden an, daß Sie sich für pdf-Dateien entschieden haben.

- `latex` verlangt Bilder im PostScript-Format.
- `pdflatex` verlangt Bilder im png-Format.
png ist ein freies Bildformat, verlustfrei komprimiert, bis 24 Millionen Farben.

10.1 Konvertieren von gif und jpeg nach.png

10.1.1 Auf der Kommandozeile

```
giftopnm filename.gif > filename.ppm    # erzeugt filename.ppm
pnmtopng filename.ppm > filename.png    # gotcha!

djpeg -pnm filename.jpeg                # erzeugt filename.ppm
      # oder filename.pgm, wenn's ein graustufenbild war
pnmtopng filename.ppm > filename.png    # gotcha!
```

10.1.2 Mit ImageMagick

Öffnen Sie das fragliche Bild unter X mit `display filename.whatever` und klicken Sie sich durch die `Save as ... Format ...`-Fenster.

10.2 Einbinden

Zunächst brauchen Sie das package `graphicx`:

```
\usepackage{graphicx}    % in der Pr"aambel, also zwischen
                        % \documentclass und \begin{document}
```

Dann brauchen Sie ein Float (s. Kapitel Tabellen):

```
\begin{figure}[htbp]    % htbp ist LaTeX Standard-Einstellung
  \includegraphics[width=0.3\textwidth]{serielle.png}
  \includegraphics[width=3cm, height=1in]{serielle.png}
  \includegraphics[width=2in, height=1cm]{serielle.png}
  \caption{Serielle Anschl"usse an einem PC}
\end{figure}
```

10.3 Und so sieht's aus

Ich füge das Bild 3mal ein, damit Sie sehen, wie `width` und `height` funktionieren.

Siehe Bild "Serielle Anschlüsse an einem PC".

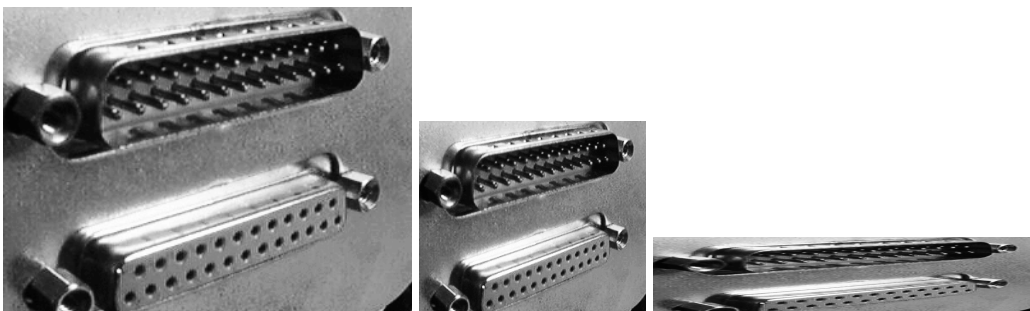


Abbildung 1: Serielle Anschlüsse an einem PC

11 Arbeitsblätter einbinden

Ist eigentlich ganz einfach, macht nur viel Arbeit:

1. Arbeitsblatt erstellen.
2. Arbeitsblatt auf einem PostScript-Drucker drucken, Ausgabe in Datei umleiten.

Unter Linux heißt diese Datei automatisch filename.ps.

Unter windos benutzt man zB den Apple LaserWriter, der macht halbwegs gutes Postscript. Die Datei heißt anything.prn, wir benennen sie um nach filename.ps.

3. Durch ghostscript und pstopnm jagen:

```
ghostscript -g600x800 -sDEVICE=ppmraw -sOutputFile=filename.ppm filename.ps
pstopnm filename.ps > filename.png
# hier mu"s man <quit> <enter> eingeben.
```

4. ... und fertig ist ein png-Bild, das man, wie gehabt, in LaTeX einbinden kann.

12 Happy LaTeXing!

Hier noch ein paar Befehle, die Sie unbedingt im Manual nachschlagen wollen:

clearpage input hrule paragraph medskip bigskip centering cite flushbottom footnotesize large Large huge
 HUGE frenchspacing em ldots onecolumn twocolumn pagestyle rm sf tt thanks vfill hfill
 array description enumerate itemize quote quotation theorem verbatim verb verse

- clearpage
- flushbottom
- ldots
- vfill
- theorem
- input
- footnotesize
- onecolumn
- hfill
- verbatim
- hrule
- large
- twocolumn
- array
- verb
- paragraph
- Large
- pagestyle
- description
- verse
- medskip
- huge
- rm
- enumerate
- bigskip
- HUGE
- sf
- itemize
- centering
- frenchspacing
- tt
- quote
- cite
- em
- thanks
- quotation

13 Anhang 1 : Kommentiertes Listing

```

% Aller Text hinter einem % wird ignoriert. % ist das Kommentar-Zeichen.
% Wer wirklich ein Prozent-Zeichen haben will, benutzt \%.

\documentclass [11pt, a4paper] {article}
% Ich habe article gew"ahlt, damit die Seitenzahlen immer rechts oben
% auf der Seite stehen. Weitere Optionen sind: book, report, letter,
% slide.
% 11 pt ist gut aus 1 m Entfernung zu lesen, zB wenn mein
% Unterrichtsskript auf dem Pult liegt.

\usepackage {german}           % aktiviert deutsche Silbentrennung
\usepackage {umlaut}           % umlaut erm"oglicht die Eingabe deutscher
% Umlaute, entweder als Buchstabe oder so: "o oder so: \"o.
% citroen schreibt man citro\"en.
\usepackage {palatino}         % meine Lieblingsschrift
\usepackage {graphicx}         % zum Einbinden von Bildern
\usepackage [margin=13mm] {geometry} % Setzen von Seitenr"andern
\usepackage {boxedminipage}    % f"ur Hervorhebungen, zB Lehrs"atze. s.u.

\author {http://www.freewebs.com/gtslehrer}
% Hier wird die interne Variable Autor gesetzt.
\date {\today}
% Hier wird das Datum gesetzt. \date ist eine eingebaute Variable, die
% das aktuelle Datum einsetzt. Wenn wir das Package german verwenden,
% wird sie in deutsch formatiert.
\title {Latex f"ur Lehrer - Test 1}

\pagestyle {headings}
% In den Kopfzeilen will ich die Seitennummer und die Kapitel"uberschrift.
\parindent 0.mm
% Den Anfang von Abs"atzen nicht (um 0mm) einr"ucken. LaTeX fri"st fast
% jede L"angeneinheit, zB pt, mm, in.
\parskip=1mm % Nach Absatz 1mm frei.
\fbboxsep=5mm % Framebox 5mm entfernt vom eingeschlossenen Text zeichnen.
\fbboxrule=0.5mm % Framebox-Rahmendicke

% Hervorhebungen machen manche Leute als Randnotizen. Quatsch.
% Hervorhebungen im Text macht man mit {\em hervorgehobene Worte}.
% \em wie emphasized=hervorgehoben.
% Hervorgehobene S"atze bekommen einen dicken Rahmen:
% ein zentriertes St"uck Text, genauso breit wie der restliche Text,
% fett (BoldFace) und etwas gr"o"ser.
\newenvironment{lehr}{
  \begin{center}
  \begin{boxedminipage}{\textwidth}
\begin{bf}
\begin{large}
}{
\end{large}
\end{bf}
\end{boxedminipage}
\end{center}
\end{lehr}

```

```
\end{bf}
\end{boxedminipage}
\end{center}
}

\begin{document}          % bis jetzt haben wir nur Verabredungen getroffen,
% die das ganze Dokument betreffen. Jetzt kommt das eigentliche Dokument.

\maketitle                % hallo latex, mach einen Titel.

\tableofcontents          % hallo latex, das Inhaltsverzeichnis kommt hierhin.

\section{Einleitung}     % Numerierung macht Latex automatisch richtig.

bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla

\section{Durchführung}

bla bla bla bla bla bla bla bla Ich fahre keinen Citroën.

\begin{lehr}
  Referentielle Integrität ist...
\end{lehr}

\section{Zusammenfassung}

bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla

\end{document}          % hallo Latex, Dokument ist hier zu Ende.
```

Latex für Lehrer - Test 1

<http://www.freewebs.com/gtslehrer>

31. Dezember 2003

Inhaltsverzeichnis

1	Einleitung	1
2	Durchführung	1
3	Zusammenfassung	1

1 Einleitung

bla bla bla bla bla bla bla bla bla bla bla bla

2 Durchführung

bla bla bla bla bla bla bla bla Ich fahre keinen Citroën.

Referentielle Integrität ist...

3 Zusammenfassung

bla bla bla bla bla bla bla bla bla bla bla bla

14 Anhang 2 : Skripten

Sie haben RedHat Linux 6.2 installiert und wollen damit komfortabel und professionell Teste erstellen? Wahlweise als pdf oder html? Dann lesen Sie weiter.

Es wird auch mit anderen Distributionen funktionieren, aber wie immer bei Linux: Your Mileage May Vary.

14.1 Verzeichnisstruktur und Prüfung

1. Ich nehme an, Sie arbeiten als Benutzer me im Home-Verzeichnis /home/me am Projekt l4l.
2. Legen Sie den Folder /home/me/Latex/l4l an.
3. Legen Sie /home/me/Latex/header1.tex an:

```
\documentclass [11pt, a4paper] {article}

\usepackage {german}
\usepackage {umlaut}
\usepackage {palatino}
%\renewcommand{\rmdefault}{phv}
\usepackage {graphicx}
\usepackage [margin=14mm] {geometry}
\usepackage {boxedminipage}
\pagestyle {headings}
\parindent 0.mm
\parskip=1mm
\fbboxsep=5mm
\fbboxrule=0.5mm

\newenvironment{lehr}{
  \begin{center}
  \begin{boxedminipage}{\textwidth}
  \begin{bf}
  \begin{large}
}{
  \end{large}
  \end{bf}
  \end{boxedminipage}
  \end{center}
}
```

4. Legen Sie /home/me/Latex/header2.tex an:

```
\author {http://your.website.goes/here}
\date   {\today}

\begin{document}
\maketitle
\tableofcontents
\vfill
```

```
\framebox{\parbox{0.9\textwidth}{
  \centering
  \large{
    {\bf {\Large Disclaimer}}
  }
}
```

Alle Informationen in diesem Dokument sind falsch, unvollständig, irreführend, irrelevant und / oder funktionieren einfach nicht.

Wenn Sie es trotzdem benutzen, und es geht dabei etwas kaputt, ist das Ihr Problem, nicht meins.

```
}
}
}
```

5. Legen Sie in 141 einen Soft-Link darauf an:

```
cd /home/me/latex/141
ln -s ../header1.tex
ln -s ../header2.tex
```

So haben Sie Ihre Hauptvorlage, und alle Ihre Dokumente sehen gleich aus.

6. Legen Sie /home/me/Latex/141/141.tex an:

```
\include {header1}          % das .tex wird automatisch erzeugt
\title {Latex f"ur Lehrer}
\include {header2}

\include{intro}

\end{document}
```

7. Die Kapitel binden Sie mit `include` ein. Das beschert u.a. einen Seitenwechsel vor jedem Kapitel. Ein absichtlicher Seitenwechsel (zB vor der Seite Übungen) wird mit `clearpage` erzeugt.

8. Legen Sie `intro.tex` an:

```
\section{Einleitung}
Standard-Textprozessoren haben gro"se Nachteile:
\begin{itemize}
\item sie fressen viel Speicherplatz;
\item sie machen vieles automatisch falsch;
\item sie zwingen zum Aufbewahren alter Versionen (falls die
soeben gespeicherte Datei, zB durch einen Absturz,
unbrauchbar wird);
\item die Platzierung von Bildern erfolgt ''auf der Seite festgenagelt''
oder so ungl"ucklich, da"s viele halbleere Seiten entstehen;
\item die Dokumente sehen auf jedem Drucker anders aus.
\end{itemize}
```

9. Lassen Sie dies durch `latex` laufen, und gucken Sie sich das Ergebnis mit `xdvi` an:

```

latex $(basename ($pwd))
startx
xdvi $(basename ($pwd)).dvi

```

10. Wenn's bis hier klappt, haben Sie die Hälfte.

14.2 Ein paar Skripten

14.2.1 Skripting

Linux ist kommandozeilenorientiert, d.h. Sie müssen Befehle eintippen. Damit das nicht immer wieder passiert, können Sie Skripten schreiben.

Versuchen Sie dies:

```

#!/bin/sh
## this ist /home/me/test1.sh

ls -la
cd /
ls -la
cd ~          # ce de leer tilde - wechselt ins homeverzeichnis

```

Machen Sie es ausführbar, und führen Sie es aus:

```

chmod 755 test1.sh      # lesbar, schreibbar, ausführbar für owner
                        # lesbar und ausführbar für den rest der welt
./test1.sh             # das aktuelle verzeichnis heißt . und ist
                        # standardmäßig nicht im PATH

```

Schreiben und testen Sie Skripten als me an Kopien der Original-Dateien. Wenn alles klappt, werden Sie root und kopieren die Skripten nach /usr/bin. Dann dürfen Sie auch den ./ weglassen.

14.2.2 Bilder einbinden

Sie möchten ein .gif einbinden? Schreiben Sie /usr/bin/giftopng :

```

#!/bin/sh
# giftopng: macht aus allen gif je ein png

if ls *.gif 2> /dev/null ; then
    for i in *.gif ; do
        j=$(echo $i | sed s/.gif/.png/)
        giftopnm $i | pnmtopng > $j
        echo $j created
    done
fi

```

Dasselbe können Sie auch mit .jpg, .fig oder .ps machen.

14.2.3 Die große Sause

```
#!/bin/sh
# /usr/bin/P # macht genau, was drinsteht

giftopng
jgptopng
figtopng
ps2png
pdflatex ($basename $(pwd)) # 3 runs needed for correct
pdflatex ($basename $(pwd)) # table of contents

xpdf ($basename $(pwd)).pdf
```

So brauchen Sie nur noch im Verzeichnis /home/me/Latex/141 P zu drücken, der Rest läuft automatisch.

14.2.4 .fig braucht eine spezielle Behandlung

Mein Drucker druckt nur mit 150 dpi, sonst ist er zu langsam.

Also habe ich eine Seitenbreite von $(210\text{mm} - 40\text{mm Rand}) / (25,4\text{mm}/\text{inch}) * 150 \text{ dots}/\text{inch} = 1004 \text{ dots}$ Seitenbreite.

Bei Bitmaps > 1004 dots verschluckt der Druckertreiber deshalb manchmal vertikale Linien in .fig-Dateien.

Also skaliere ich wie folgt:

```
#!/bin/sh

desired=900 # manche bilder sind kleiner als \textwidth

if [ -z $1 ] ; then
  echo "must give a .fig file as input"
  exit 1;
else
  file=$(basename $1 .fig)
  # echo "working on $file.fig..."
  fig2dev -L ppm $file.fig $file.ppm
  width=$(head -3 $file.ppm | tail -1 | cut -f1 -d" ")
  rm -f $file.ppm
  echo -n "width of $file.fig is $width ... "
  mag=$(echo "$desired/$width" | bc -lq)
  # echo "mag = $mag"
  fig2dev -L png -m $mag $file.fig $file.png
fi
```