LaTeX für Fortgeschrittene

Dr. Wolfgang Riedel, TU Chemnitz, URZ

Zi. 1/B301b, Tel.: 31422, E-Mail: w.riedel@hrz

Stand: 23. Dezember 2011

Inhaltsübersicht

- Vorausgesetzte Grundlagen
- Seitenlayout
- KOMA-Script
- Strukturen für komplexe Dokumente
- Verzeichnisse
- Mikrotypografie
- Grafiken und Bilder
- Zeichensätze
- Hypertext
- Präsentationen
- System-TeXnisches
- Schnittstellen zu anderer Software
- Was fehlt noch?

1. Vorausgesetzte Grundlagen

- Grundprinzip von TeX: Programmiersprache
  für die Anwendung bedeutet das: 3 Schritte

- Formatierer: was erzeugen sie?
  `latex`
  `pdflatex`
- Dokumentklassen: was ist das?
  `scrartcl`, `scrreprt`, `scrbook`, `scrlettr2`
- Klassenoptionen: was ist das?
  `10pt` ... `12pt`, `twoside`, `a4paper`
- Formatierung: Blocksatz / Flattersatz
- Hervorheben von Text: `emph`
- Schriftgrößen: `tiny` ... `Huge`
- Schriftschnitte: `textsf`/`sffamily` ...
- Präambel: was steht dort üblicherweise?
- Zeichensatzauswahl: Zusatzpaket `fontenc`
- Eingabekodierung: Zusatzpaket `inputenc`
- Spracheinstellung: Zusatzpaket `ngerman`
- Textstrukturen
  - Kapitel, Abschnitte, Überschriften: `part` ... `section` ... `subparagraph`
  - Textformatierung linksbündig, rechtsbündig, zentriert: `flushleft`, `flushright`, `center`
  - Hervorhebung (Einrückungen): `quote`
  - unformatierter Text: `verbatim`, `verb`
  - Listen: `itemize`, `enumerate`, `description`
- Einbinden von Grafiken und Bildern: Zusatzpaket `graphicx`, `includegraphics`
- Tabellen
  - `tabbing`, `tabular`
  - Spaltenformatierung: `l`, `r`, `c`, `p`, `|`, `@`, `multicolumn`
  - gleitende Tabellen: `table`, `caption`, `ref`
- Formeln
  - eingebettete und abgesetzte Formeln: `$`, `equation`
  - Elemente in Formeln
  - Schriftarten in Formeln

- ➢ Standardfunktionen
- ➢ mathematische Operatoren
- ➢ Klammern
- ➢ Mehrzeilige Formeln
- ➢ Akzente, Pfeile, Sonderzeichen

2. Seitenlayout

2.1 Seitenstil

`\pagestyle{`*stil*`}`
`\thispagestyle{`*stil*`}`

	Seitenkopf	Seitenfuß
`plain`	leer	Seitennummer zentriert
`empty`	leer	leer
`heading`	Seitennummer, Überschriftsinformation entspr. Dokumentenstil	leer
`myheadings`	Inhalt von `leftmark` und `rightmark`	leer

Standard: `plain`

`leftmark` und `rightmark`

- ➤ Variable, die die aktuellen „Werte“ für linken und rechten Seitenkopf enthalten
- ➤ werden durch (interne) Befehle `\markboth` und `\markright` belegt
- ➤ diese wiederum werden bei der Definition von Gliederungsbefehlen benutzt (`\section` ...)

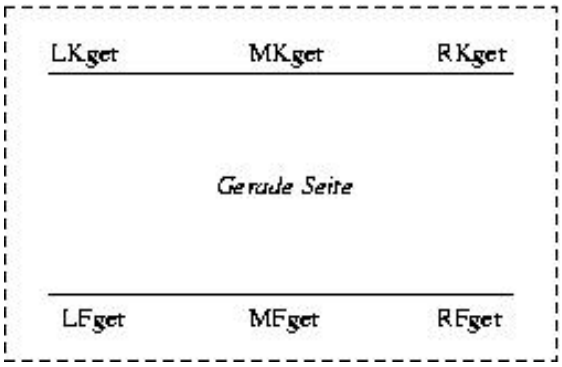
Zusatzpaket `fancyheadings`

- ➤ dreiteilige Kopf- und Fußzeilen
- ➤ Linien in Kopf- und Fußzeilen

- ➤ Mehrzeilige Kopf- und Fußzeilen
- ➤ ...

Aktivierung: `\pagestyle{fancy}`

`\lhead[`*LKger*`]{`*LKunger*`}` `\lfoot[`*LFger*`]{`*LKunger*`}`
`\chead[`*MKger*`]{`*MKunger*`}` `\cfoot[`*MFger*`]{`*MKunger*`}`
`\rhead[`*RKger*`]{`*RKunger*`}` `\rfoot[`*RFger*`]{`*RKunger*`}`




```
\lfoot{\tiny\sf \LaTeX\ II: \rightmark}
\cfoot{}
\rfoot{\tiny\sf \theslide{} (\pageref{Lastpage})}
```

2.2 Seitennummerierung

`\pagenumbering{`*stil*`}`

`arabic`	(arabische) Zahlen
`roman`	kleine römische Zahlen
`Roman`	große römische Zahlen
`alph`	kleine Buchstaben
`Alph`	große Buchstaben

Benutzen des Seitenzählers: `\thepage`
Setzen des Seitenzählers: `\setcounter{page}{`*zahl*`}`

2.3 Seitenformat

Standard: Papiergröße (`letter`), Hochformat (`portrait`)

A4-Papier: Klassenoption `a4paper`

Querformat: Klassenoption `landscape`

Seitenformat einstellbar:

- LaTeX: `\usepackage[dvips]{geometry}`
  oder als Option bei `dvips`: `-t landscape`
- PDFLaTeX: `\usepackage[pdftex]{geometry}`

2.4 Abstände, Maße

Vertikale Abstände:

`\bigskip`
(`\smallskip`, `\medskip`)
`\vspace{`*mass*`}`, `\vspace*{`*mass*`}`
`\vfill`, `\vfill*`

Horizontale Abstände:

`\quad`, `\qquad`,
`\hspace{`*mass*`}`, `\hspace*{`*mass*`}`
`\hfill`, `\hfill*`

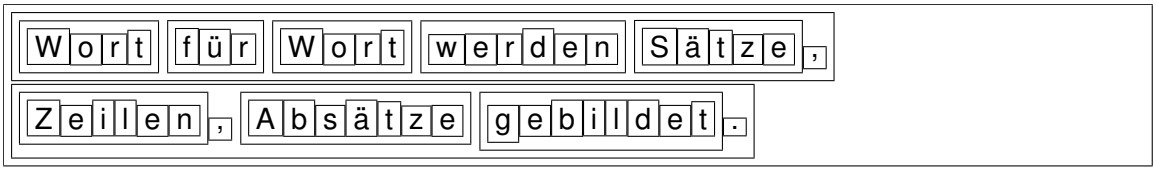
Maßeinheiten für Längenangaben:

`mm`	Millimeter
`cm`	Zentimeter = 10 mm
`in`	Inch $\approx$ 25 mm
`pt`	Point $\approx \frac{1}{72}$ in $\approx \frac{1}{3}$ mm
`em`	proportional zur Zeichenbreite des aktuellen Fonts
`ex`	proportional zur Zeichenhöhe des aktuellen Fonts

2.5 Boxen

Erinnerung:

Formatierung ist immer das Anordnen von Boxen



2.5.1 LR-Boxen

- ➤ Anordnung des Inhalts von *links nach rechts*
- ➤ kein Zeilenumbruch!

`\mbox{`*text*`}`
`\makebox[`*breite*`] [`*position*`]{`*text*`}`
`\fbox{`*text*`}`
`\framebox[`*breite*`] [`*position*`]{`*text*`}`

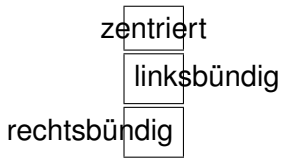
position

`l` Text erscheint linksbündig in der Box
`r` Text erscheint rechtsbündig in der Box

ohne Parameter: Text erscheint zentriert in der Box

breite kann breiter oder schmaler als wirkliche Textbreite sein

```
\framebox[8mm]{zentriert} \\
\framebox[8mm][l]{linksbündig} \\
\framebox[8mm][r]{rechtsbündig}
```



Vertikale Verschiebung von LR-Boxen

`\raisebox{`*lift*`}{`*text*`}`

```
Grundlinie \raisebox{1ex}{hoch} und
\raisebox{-1ex}{tief} und zurück
```

Grundlinie $^{\text{hoch}}$ und $_{\text{tief}}$ und zurück

2.5.2 Absatzboxen

erzeugen Absätze, d.h. mit automatischem Zeilenumbruch

\parbox[*position*]{*breite*}{*text*}
\begin{minipage}[*position*]{*breite*}
text
\end{minipage}

position: Ausrichtung in Bezug auf die laufende Zeile

- `b` unterste Zeile der Absatzbox auf gleicher Höhe
- `t` oberste Zeile der Absatzbox auf gleicher Höhe

ohne Parameter: Absatzbox ist vertikal zentriert

```
- \begin{minipage}[b]{3cm}
das ist aber auch ein schöner Text
\end{minipage}\hfill
\begin{minipage}{2cm}
das ist ein schöner Text
\end{minipage}\hfill
\begin{minipage}[t]{3cm}
das ist aber auch ein schöner Text
\end{minipage} -
```

das ist aber auch - ein schöner Text

das ist ein schöner Text

das ist aber auch - ein schöner Text

2.5.3 Rule-Boxen

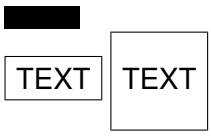
erzeugt gefülltes Rechteck

\rule[*lift*]{*breite*}{*höhe*}

besonders interessant: Ruleboxen der Breite 0

```
\rule{10mm}{3mm}

\fbox{TEXT}
\fbox{\rule[-4mm]{0mm}{10mm}TEXT}
```



2.5.4 Zusatzpaket `fancybox`

Benutzung von Zierrahmen

`\shadowbox`{text}	schattierte Box
`\doublebox`{text}	Box mit Doppelrahmen
`\ovalbox`{text}	Box mit abgerundeten Ecken
`\Ovalbox`{text}	wie `ovalbox`, dickerer Rand

```
\shadowbox{Eine schattierte Box}
\doublebox{Eine Doppelbox}
\ovalbox{Eine ovale Box}
\Ovalbox{Eine Ovale Box}
```

2.5.5 Skalierte Objekte

`\scalebox`{*h-faktor*}[*v-faktor*]{*objekt*}
`\resizebox`{*breite*}{*hoehe*}{*objekt*}
`\reflectbox`{*text*}

```
groß und g\scalebox{2}{r}
\scalebox{4}{ö}\scalebox{6}{s}\scalebox{8}{s}
\scalebox{10}{e}\scalebox{12}{r}
```

groß und gröSSer

```
\fbox{\resizebox{3cm}{3cm}{\textbf{MORD}}
in der Badewanne}
```

```
\reflectbox{gespiegelt}
```

ʇlǝpǝiqsǝƃ

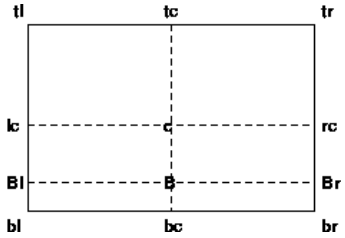
2.5.6 Gedrehte Objekte

Befehl aus dem Zusatzpaket `graphicx`

`\rotatebox[`*param*`]{`*winkel*`}{`*objekt*`}`

Parameter:

- `x=`*mass*, `y=`*mass*: Verschiebung des Drehpunkts
- `origin=`*label* (Standard: `Bl`)




```
Jetzt geht es \rotatebox{25}{bergauf}
und \rotatebox[origin=Br]{-20}{bergab}
... \rotatebox{-20}{bergab}!
```

Jetzt geht es bergauf und bergab ... bergab !

Andere Möglichkeit zum Drehen: Zusatzpaket `rotating`

```
\begin{rotate}{winkel}
...
\end{rotate}
```

```
\begin{sideways}
...
\end{sideways}
```

`sideways` = `rotate` mit 90°

```
Das ist der Text, der nicht gedreht wird. \\
\begin{rotate}{15}
\fbox{Das ist der Text, der gedreht wurde.}
\end{rotate} \\
Das ist der Text, der nicht gedreht wird.
```

Das ist der Text, der gedreht wurde.

Das ist der Text, der nicht gedreht wird.

Das ist der Text, der nicht gedreht wird.

Besonderheit der `rotate`-Umgebung: zusätzlicher vertikaler Platz wird nicht berücksichtigt!

Beispiel für `sideways`: Tabelle mit vielen Spalten, für den Tabellenkopf ist nicht genug Platz ...

```
\begin{tabular}{l|l||l ...
Titel & uid &
   \begin{sideways}{\tiny sun4}\end{sideways} &
   \begin{sideways}{\tiny sun5}\end{sideways} &
   \begin{sideways}{\tiny NeXT}\end{sideways} &
   ...
```

Ergebnis

3. KOMA-Script

3.1 Allgemeines

- Neuentwicklung der Standard-LaTeX-Dokumentklassen `scrartcl`, `scrreprt`, `scrbook` – kein „Ersatz“
- Anpassung an europäisches Papierformat und Layout-Gewohnheiten
- funktionelle Erweiterungen, . . .

3.2 Satzspiegel (traditionell)

Viele (auf einander abgestimmte) interne Variable bestimmen das Layout:

- Seitenhöhe, -breite
- Höhe und Breite des Textkörpers
- Spaltenbreite und -abstand bei Mehrspaltensatz
- Randbreiten links und rechts, oben und unten
- Höhe der Kopf- bzw. Fußzeile, Abstand zum Textkörper
- . . .

Übersicht

Alle diese Maße sind tabu!

Modifikationen nur mittels:

- Zusatzpaket `parskip`
  kein Erstzeileneinzug bei Absatzanfang, dafür größerer Absatzabstand
- Zusatzpaket `setspace`
  definiert 2 Kommandos für Zeilenabstand
  `\doublespacing`
  `\onehalfspacing`
- Kommando `\linespread{`*zahl*`}`

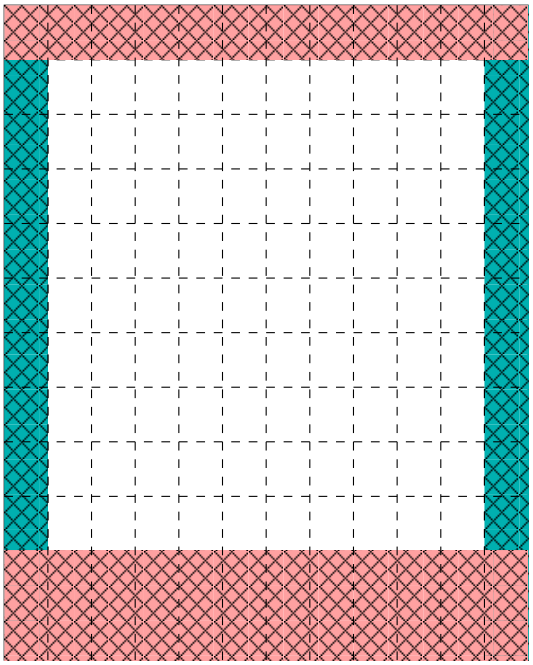
```
\linespread{1.08}
```

- Zusatzpakete `mdwlist`, `paralist`, `expdlist`
  anderes Listenlayout

Bei KOMA-Skript:

Satzspiegel durch *n-Teilung* (nach Tschichold)
die Seite wird senkrecht und waagerecht in jeweils *n* gleiche Streifen zerlegt

oberer Rand ← 1 Streifen ($= \frac{1}{n}$ Höhe)
unterer Rand ← 2 Streifen
linker Rand ← 1 Streifen
rechter Rand ← 1 Streifen
bei doppelseitig: halber innerer Rand ← 1/2 Streifen
der „Rest“ kann gefüllt werden (Text-, Kopf-, Fußbereich, . . .)



n=12

je größer *n*, desto kleiner die Ränder und desto größer der Textbereich!

Steuerung durch Klassenoption: `DIV`*n*

Bindekorrektur: `BCOR`*m*

eingestellte Standardwerte:

Grundschriftgröße	`10pt`	`11pt`	`12pt`
n	8	10	12
m	0	0	0

3.3 Klassenoptionen

Papierformat

- `letterpaper`, `legalpaper`, `executivpaper`
- a*X*`paper`, b*X*`paper`, c*X*`paper`, d*X*`paper`
- `isopaper[`*reihe*`]{`*formatnummer*`}`

Layout

- `openany`, `openright`: Beginn eines Kapitels (`=chapter`) auf jeder oder nur auf rechter Seite
- `headsepline`, `footsepline`: Trennlinie unter Seitenkopf bzw. über Seitenfuß
- `parskip` (und Varianten): Absatzabstand anstelle von Absatzeinzug

Schriftgröße

- außer `10pt`, `11pt`, `12pt` auch `8pt`, `9pt`, `14pt`, `17pt`, `20pt` (benutzt Zusatzpaket `extsizes`)
- `smallheadings`, `normalheadings`, `bigheadings`: normalerweise große Überschriften (`big`)

Umgang mit Bildern

- `draft`: Ausgabe von Rahmen anstelle der Bilder, Markierung von übervollen Zeilen durch Kästchen am Zeilenende
- `final`: wie gewohnt mit Bildern

. . . und viele andere Optionen

3.4 Steuerung der verwendeten Zeichensätze

(Grundschriftart kann im Original-LaTeX geändert werden, nicht aber die Schriftarten für spezielle Elemente wie Bildunterschriften, Kapitelüberschriften usw.)

`\setkomafont{`*element*`}{`*befehle*`}`
`\usekomafont{`*element*`}`

verantwortungsbewusst einsetzen!

element kann dabei sein:

`caption`	Unterschrift/Überschrift Abbildung oder Tabelle
`captionlabel`	Beschreibungslabel („Tabelle“)
`chapter`	Überschrift Hauptkapitel
`footnote`	Marke und Text einer Fußnote
`pagehead`	Seitenkopf und -fuß
...	

```
\setkomafont{captionlabel}
            {\usekomafont{descriptionlabel}}
```

3.5 Seitenstil

zusätzliche Steuerungsmöglichkeiten mittels

`\titlepagestyle`	für Titelseite
`\partpagestyle`	für Startseite eines `part`
`\chapterpagestyle`	für Startseite eines `chapter`
`\indexpagestyle`	für erste Indexseite

3.6 Textbausteine

zusätzliche Möglichkeiten für

- Titelei (für Bücher)
  - „Schmutztitel“
  - Titelblattrückseite
  - ...
- Gliederung, z.B.

  `\minisec{`*überschrift*`}`

  Zwischenüberschrift, ohne Gliederung, keine Abstände

➤ Listen, z.B.

\begin{labeling}[*trennzeichen*]{*längstes Muster*}

tabellenartiges Layout: zweite Spalte wird nach dem längsten Eintrag der ersten Spalte ausgerichtet

3.7 KOMA-Script-eigene Zusatzpakete

`scrpage2` definiert Seitenstil `scrheadings` (analog `fancyhdr`) und `scrplain`

`scrdate` `\todaysname` – aktueller Wochentag

`scrtime` `\thistime`[trennung] – aktuelle Uhrzeit als „Stunden *trennung* Minuten“ (Standard: :)

```
Heute ist \todaysname,
und jetzt ist es: \thistime

\LaTeX{} kann standardmäßig nur: \today
```

Heute ist Freitag, und jetzt ist es: 10:49
LaTeX kann standardmäßig nur: 23. Dezember 2011

4. Strukturen für komplexe Dokumente

4.1 Titelseite („Titelei“), Deckblatt

Festlegen der Bestandteile des Titels:

\title{*dokumententitel*}
\author{*dokumentenautor*}
\date{*erstellungsdatum*}
\thanks{*fussnote*}

Ausgabe des Titels:

\maketitle

⇒ separate Titelseite bei

- ➤ `scrreprt/report`
- ➤ `scrbook/book`
- ➤ `scrartcl/article` nur bei Option `titlepage`

Layout: ohne Seitenkopf und -fuß
Texte vertikal zentriert

```
\title{Unsere größten Abenteuer}
\author{Winnetou\\Prärie\\USA \and
        Old Shatterhand\thanks{email:
        osh@karl.may.de}\\Hohenstein-Ernstthal\\
        Germany}
\date{25.02.1892}
\maketitle
```

Ergebnis

4.2 Zusammenfassung

`\begin{abstract}`
zusammenfassungstext
`\end{abstract}`

⇒ Zentrierte Überschrift 'Abstract' / 'Zusammenfassung'

- ➤ `scrartcl`: Text beidseitig eingerückt, kleinerer Font
- ➤ `scrreprt`: extra Seite, normale Schriftgröße
- ➤ `scrbook`: —

```
\begin{abstract}
This is an example input file.  Comparing it with
the output it generates can show you how to
produce a simple document of your own.
\end{abstract}
```

Ergebnis

Besonderheit bei KOMA-Script:

`scrartcl` und `scrreprt` kennen die Option

`abstracton` bzw. `abstractoff`

bewirkt das Erzeugen der Überschrift „Zusammenfassung“

Standard: `off`

4.3 Untergliederung

Bekannt: `\section`, `\chapter`, ...

Allgemeiner:

> *gliederungsbefehl*[*kurzform*]{*überschrift*}
> *gliederungsbefehl**{*überschrift*}

Kurzform:	Inhaltsverzeichnis Seitenkopf (bei `headings`)
* :	keine Nummerierung nicht ins Inhaltsverzeichnis

Gliederungszähler werden automatisch hochgezählt

explizites Setzen:

> `\setcounter`{*zähler*}{*zahl*}

zähler: `section`, `chapter`, ...

```
\setcounter{section}{5}
```

das nächste Kapitel (=nächster Aufruf von `\section`) bekommt die Nummer 6

Besonderheit bei KOMA-Script:

Klassenoption `chapterprefix`:
Überschrift als „Kapitel *nr*“ + Newline

Standard: `nochapterprefix`

4.4 Anhang

> `\appendix`
> *anhangstext*

Wirkung: Hauptgliederungszähler wird auf 0 zurückgesetzt, Nummerierungsstil: große Buchstaben

```
\appendix
\section{Versuchsauswertung}
\subsection{Versuch: Leitfähigkeit}
  ...
\section{Formelsammlung}
```

Ergebnis

Besonderheit bei KOMA-Script:

Klassenoption `appendixprefix`:
Überschrift als „Anhang *nr*“ + Newline

Standard: `noappendixprefix`

4.5 Arbeit mit Teildokumenten

Zerlegung eines großen Dokuments in einzelne Files (z.B. kapitelweise), damit

- ➤ leichteres Editieren möglich
- ➤ Wiederverwendung von Textstücken
- ➤ ...

`\include{`*file*`}`

⇒ das File wird beim Formatieren an der betreffenden Stelle eingefügt

Zusätzlich möglich (in der Präambel!):

`\includeonly{`*fileliste*`}`

nur die `\include`-Anweisungen werden wirksam, deren Filename in der Fileliste stehen

```
\includeonly{kap1,kap3}
\begin{document}
\include{kap1}
\include{kap2}
\include{kap3}
\end{document}
```

Alternative (TEX-Kommando):

`\input{`*file*`}`

Einfügen der Datei ohne vorhergehenden Seitenumbruch

4.6 Fußnoten, Randnotizen

Markierung am laufenden Text, die im Seitenfuß (mit Text) oder am Seitenrand erscheinen.
Bei Fußnoten: nummeriert (bei `article` fortlaufend, bei `report` und `book` kapitelweise; kleinere Schriftgröße)

`\footnote{`*text*`}`
`\marginpar{`*markierung*`}`

```
Eine häufig anzutreffende Form
\marginpar{\rule[-13.5mm]{1mm}{16mm}}
für eine Randnotiz ist ein vertikaler Balken,
mit dem Textpassagen gekennzeichnet werden
Diese Markierung\footnote{Das ist eine Fußnote}
macht Textänderungen oder -ergänzungen gegenüber
vorhergehenden Versionen kenntlich.
Man kann auch so \marginpar{$\Longrightarrow$}
markieren.
```

Ergebnis

4.7 Nützliche Zusatzpakete für Textsatz

4.7.1 `verbatim`-Paket

- Erweiterung der `verbatim`-Umgebung (Bugbeseitigung)
- zusätzliches Kommando

  `\verbatiminput{`*filename*`}`

hier im Skript sind Beispiele immer:
`\verbatiminput{bsp_nr}`
`\input{bsp_nr}`

Vorteile:

- macht LaTeX-Quelltext aufgeräumter
- ermöglicht externe Beispieltests (`make`)

weitere interessante Pakete: `moreverb`, `fancyvrb`

4.7.2 Darstellung von Quellcode: `listings`-Paket

- `verbatim` ist für Quelltextdarstellungen unbefriedigend
- neue Umgebung: `lstlisting`

```
\begin{lstlisting}[schluessel=wert,...]
quelltext
\end{lstlisting}
```

- Schlüssel-Wert-Paare definieren die Formatierung
  - `language`: Java, C++, Python, PHP, XML ...
  - `basicstyle`: `\sffamily\color{Navy}\footnotesize`
  - `keywordstyle`
  - `commentstyle`
  - `stringstyle`
  - `numbers`: left
  - `numberstyle`
  - ...

Ein Beispiel:

```
\begin{lstlisting}[language=C,
 basicstyle=\rmfamily, keywordstyle=\bfseries,
 frameround=tttt, frame=single, framesep=6pt]
main(argc, argv)
int argc; char *argv[];
{
    /* Das ist ein Test */
    printf("Hello listings\n");
}
\end{lstlisting}
```

```
main(argc, argv)
int argc; char *argv[];
{
    /* Das ist ein Test */
    printf("Hello listings\n");
}
```

Weitere Beispiele

4.7.3 `multicol`-Paket

- die LaTeX-Klassen kennen maximal 2 Textspalten (`twocolumn`)

➤ neue Umgebung:

\begin{multicols}{*zahl*}[*überschrift*]
text
\end{multicols}

```
\begin{multicols}{3}[\section*{\TeX--Tagung
DANTE '93 in Chemnitz}]
4 Tage lang trafen sich ca.~100 Mitglieder der
"`Deutschsprachigen Anwendervereinigung
\TeX\ e.V."', um über Fragen und Probleme der
Anwendung von \TeX\ und \LaTeX\ Vorträge zu ...
```

Ergebnis: (1) „löchriger Satz“
(2) „Flattersatz“ durch \raggedright
(3) Flattersatz mit Silbentrennung durch \RaggedRight
Zusatzpaket ragged2e

4.7.4 endnotes-Paket

bei Belletristik sind Fußnoten unüblich, dafür „Endnoten“: erläuternde Texte am Ende des Dokuments

➤ Zusatzpaket endnotes

➤ in der Präambel: \let\footnote=\endnote

➤ an gewünschter Stelle: \theendnotes

5. Verzeichnisse

5.1 Inhaltsverzeichnis

wird automatisch erzeugt, muss aber explizit ausgegeben werden:

```
\tableofcontents
```

Ein Beispiel

Textverschiebungen beachten!

analoge Verfahrensweise für Bildverzeichnis (`\listoffigures`) und Tabellenverzeichnis (`\listoftable`)

Zusätzliche, explizite Einträge in Verzeichnisse:
`\addcontentsline`, `\addtocontents`

Nutzerspezifische Verzeichnisse möglich

5.2 Stichwortverzeichnis / Index

- Sammlung von Stichworten mit der/den Seitenzahl(en) ihres Auftretens
- Vorbereitung zur Erzeugung eines Index:

  `\usepackage{makeidx}`
  `\makeindex` (ebenfalls in der Präambel)

- Definition eines Stichworts:

  `\index{`begriff`}`
  `\index{`begriff`!`unterbegriff`}`

- im `index`-Kommando weiterhin möglich:
  - Festlegen von Seitenbereichen
  - Steuerung der Darstellungsform
  - Sonderbehandlung von Sonderzeichen
- Ausgabe des Stichwortverzeichnis:

  `\printindex`

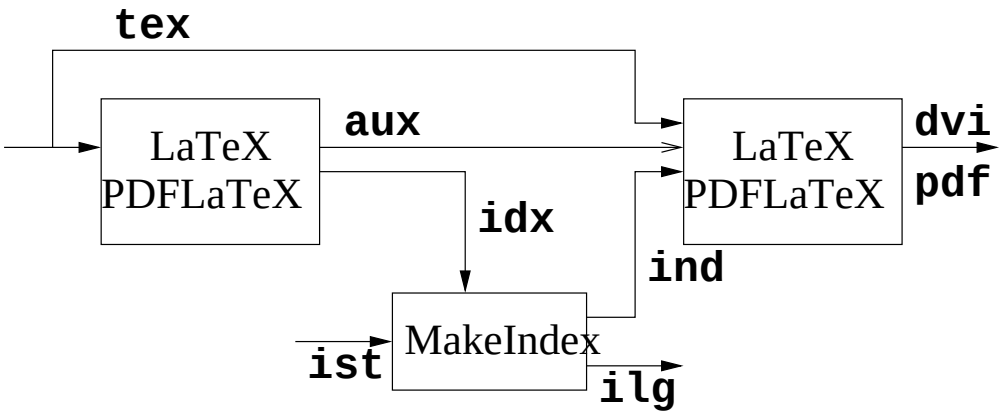
Ein Beispiel:

```
Ich esse gern Obst\index{Obst},
besonders liebe ich die Banane\index{Obst!Banane}
und die Orange\index{Obst!Orange}.
```

Ergebnis

- Formatierung erzeugt: `file.idx`
- fortlaufende Liste von Wörtern mit der betreffenden Seitenzahl

- Erzeugung der benötigten alphabetisch sortierten Liste mittels externer Indexgeneratoren
  - MakeIndex
  - Xindy



Ergebnis:

```
\begin{theindex}
\item stichwort seite
    \subitem unterstichwort seite
...
\end{theindex}
```

Danach erneute Formatierung des (Text-)Dokuments

Aufruf:

`makeindex` [*optionen*] [`-s` *stil*] *idxfile*

Optionen:

`-g` deutsche Sortierreihenfolge

...

stil: standardmäßig `gind.ist`

Reales Beispiel

Zusatzpakete

- `showidx` gibt `\index`-Befehle am Seitenrand aus (zur Fehlersuche)
- `repeatindex` Seitenumbrüche intelligent steuern
- `tocbibind` Auflistung im Inhaltsverzeichnis

- ➤ `indxcite` erzeugt automatisch Autorenverzeichnis
- ➤ `splitidx` Erzeugung mehrerer Indexe

5.3 Literaturverzeichnis

- ➤ Zitate müssen als solche gekennzeichnet werden
- ➤ Einfügen von Referenzen im Text auf die Literaturquellen

`\cite`{bezug}
`\cite`{bezug1 ,bezug2[,...]}

Auflistung aller Literaturquellen, auf die im Text verwiesen wird: das Literaturverzeichnis

`\begin`{`thebibliography`}{mustermarke}
`\bibitem`[marke]{bezug} text
...
`\end`{`thebibliography`}

```
\begin{thebibliography}{99}
\bibitem{la} Leslie Lamport. \emph{\LaTeX\ -
   A Document Preparation System}.
   Addison-Wesley Co., Inc., Reading, MA, 1985
...
\bibitem[6a]{kn.a} Vol. A:
   \emph{The \TeX book}, 1986
...
\end{thebibliography}
```

Ergebnis

Resüme:

- ➤ ziemlich aufwändig, wenn Daten per Hand eingegeben werden!
- ➤ Einheitlichkeit nicht garantiert

deshalb andere Technologie:

- ➤ Verwendung einer separaten, „gepflegten“ Literaturdatenbank
- ➤ Sammlung aller relevanten Literatur (einer Arbeitsgruppe)
- ➤ Verwaltung solcher Datenbanken durch verschiedene Softwareprodukte möglich

„BIBTEX-Format“:

```
@book{la,
    author = {Leslie Lamport},
    title  = {LaTeX - A Document Preparation System},
    publisher = {Addison-Wesley Co., Inc.},
    address   = {Reading, MA},
    year      = {1985}}
...
```

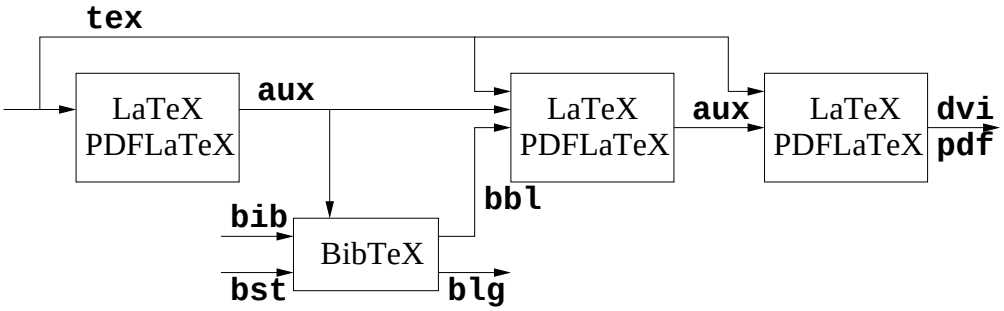
Eintragstypen:
`article`, `book`, `booklet`, `conference`, `inbook`, `incollection`, `inproceedings`, `manual`, `masterthesis`, `misc`, `pdthesis`, `proceedings`, `techreport`, `unpublished`

Felder:
`address`, `annote`, `author`, `booktitle`, `chapter`, `crossref`, `edition`, `editor`, `howpublished`, `institution`, `isbn`, `issn`, `journal`, `key`, `language`, `month`, `note`, `number`, `organization`, `pages`, `publisher`, `school`, `series`, `title`, `type`, `url`, `volume`, `year`

Aufgabe:

- ➤ Ermittlung der *notwendigen* Literaturstellen aus einer Datenbank
- ➤ Sortierung
- ➤ Erzeugung der `thebibliography`-Notation

⇒ externes Zusatzwerkzeug BiBTEX



Reales Beispiel

Im LATEX-Dokument:

`\bibliographystyle{`stil`}`
`\bibliography{`datenbank`}`

Vorhandene Stile:

`plain`	Alphabetische Sortierung der Einträge, numerische Marken – *Standard*
`unstr`	Ausgabe in der Reihenfolge der Zitierung, numerische Marken
`alpha`	Alphabetische Sortierung, Marken aus Autorname und Erscheinungsjahr
`abbrv`	wie `plain`, aber Abkürzung von Vornamen, Monaten, Zeitschriftennamen
`plaindin`	wie `plain`, gemäß der Zitiernorm DIN 1505 Teil 2
...	

Stile können selbst generiert werden: `custom-bib`-Paket

```
tex makebst
```

- „Dialogprogramm“, stellt eine Vielzahl von Fragen
- erzeugt individuell zugeschnittenes `bst`-File

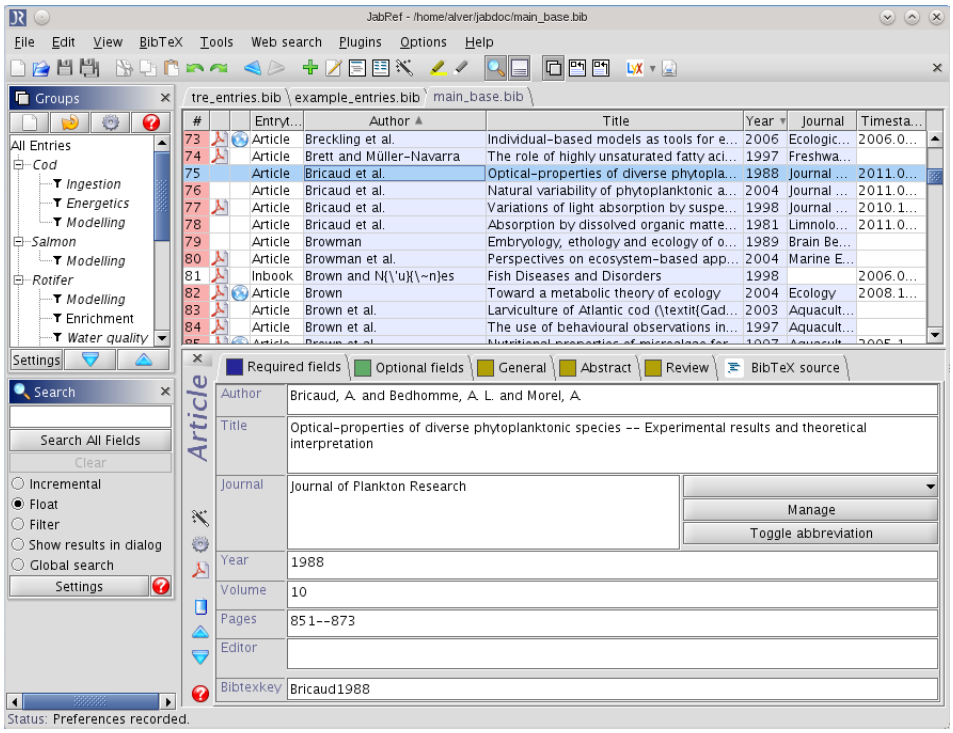
Zusatzpakete

- `natbib` „NAturwissenschaftliche BIBliographie“: weitere `cite`-Befehle
- `bibentry` Bibliographieeinträge im Fließtext
- `jurabib` Anpassungsfähige Kurztitelverweise
- `camel` Unterstützung für Gesetzestexte
- `chapterbib` Literaturverzeichnis kapitelweise
- `bibunits` Bibliographien für beliebige Einheiten
- `bibtopic` Nach Themen sortierte Literaturhinweise
- `multibib` Separate globale Bibliographien
- ...

Externe Literaturverwaltungssysteme:

Export-Schnittstelle zu BiBTEX

- Endnote, Reference Manager
- RefWorks (frühe Cloud-Anwendung)
- Citavi
- JabRef: Java-basiertes, freies Literaturverwaltungsprogramm
- lauffähig unter Linux, Windows, Mac und als Java Start
- `http://jabref.sourceforge.net/`



6. Mikrotypografie

Satzvorschriften für einzelne Zeilen, Wörter und Zeichen

Basis für das Setzen von Text ist der Duden!
(manchmal sind mehrere Varianten einer Schreibweise zugelassen . . .)

Informationen:

- Der große Duden
- HEINZ W. PAHLKE: Buchsatz für Autoren, Shaker Media, Aachen 2008
- MARION NEUBAUER: Feinheiten bei wissenschaftlichen Publikationen – Mikrotypografie-Regeln, Teil 1 + 2
  `http://www.dante.de/dante/DTK/`
  Hefte 4/96 und 1/97

Abkürzungen

- keine Beugungsendung, kein Plural (ggf. durch Umschreiben vermeiden):
  des Jh., des PC, die PC
- bei Plural mit Beugungsendung, um Missverständnisse zu vermeiden:
  die GmbHs, die Bde.
- Abkürzungen, die im Wortlaut gesprochen werden, mit Punkt(en) schreiben:
  z. B., Dr.
- auch Abkürzungen, die nicht mehr ausgesprochen werden:
  a. D., i. V.
- sonst ohne Punkt:
  GmbH, BGB, USA
- bei mehrteiligen Abkürzungen kleinen Zwischenraum („Spatium“, `\,`) einfügen:
  `z.\,B.` z. B.
  ebenso nach einem Abkürzungspunkt im laufenden Text

Striche

`-`	Bindestrich	O-Beine (*ohne Leerzeichen!*)
`--`	von-bis	10–18 Uhr, Paris–Dakar (*ohne LZ!*)
	Gedankenstrich	„Mein Sohn, was birgst du so bang dein Gesicht“ – „Siehst, Vater, du den Erlkönig nicht?“
	Einschub	ich habe – ohne zu überlegen – die Taste ESC gedrückt
`---`	Gedankenstrich	modish—use (*nur im Amerikanischen!*)
`$-$`	Minus	-1 (*nicht:* -1)

Satzzeichen, Sonderzeichen

- niemals Leerzeichen *vor* Satzzeichen, aber *immer* danach
  Satzzeichen, aber
- Klammern: Leerzeichen **vor öffnender** Klammer, analog **nach schließender** Klammer, aber nicht nach öffnender bzw. vor schließender Klammer
  es gab (bis vor kurzem) eine Regelung
- Auslassungszeichen („Ellipse“): `\dots` $\cdots$ Zeilenumbruch davor verhindern: `~`
- Am Satzende: kein weiterer Punkt, aber andere Satzzeichen möglich
  Bis später . . . !

Zahlen, Nummern

- Zahlen ab 10 durch Ziffern, kleinere durch Worte darstellen:
  fünf Schritte, 27 Schritte
- große Zahlen von rechts in Dreiergruppen darstellen, Punkttrennung möglich (Verwechslungsgefahr):
  `123\,456\,789, 100.999` 123 456 789, 100.999
  bei Dezimalstellen vom Komma aus gruppieren:
  `2,718\,281\,8` 2,718 281 8
- Telefon-, Fax- und Postfachnummern (nicht: Postleitzahlen) in Zweiergruppen darstellen, Ortskennzahl ebenso und in runde Klammern setzen:
  `(0\,62\,21)~2\,97\,66, +49~(62\,21)~2\,97\,66`
  (0 62 21) 2 97 66, +49 (62 21) 2 97 66

Maßeinheiten

- Zahlen mit Maßeinheiten: Abstand notwendig (mit ~ verbinden)
  `10~m` , `5~EUR` 10 m , 5 EUR
  in englischsprachigen Texten: kleiner Zwischenraum `\,`
- bei Prozent (`\%`) und Promille (`\textperthousand`) nur Spatium:
  `12\,\%` , `0.5\,\textperthousand` 12 % , 0.5 ‰
  bei Zusammensetzungen kein Zwischenraum:
  25%ige Steigung, die 5%-Hürde
- Gradzeichen (`\textdegree`) ohne Zwischenraum:
  `30\textdegree` , `17\textcelsius` 30 ° , 17 ℃
- Maßeinheiten ohne Ziffern aussschreiben:
  wenige Millisekunden
- Abkürzungen der Maßeinheiten, Himmelsrichtungen, Währungen, chemische Elemente: ohne nachfolgenden Punkt:
  m, SW, €, NaCl
- Abkürzungen von Zahlwörtern mit Punkt:
  Tsd., Mio., Mrd.

Gänsefüße

- es gibt:
  - Apostroph

- ➢ Zollzeichen (=Doppelapostroph)
- ➢ untere (öffnende) und obere (schließende) Anführungszeichen
- ➢ jeweils einfach und doppelt
- ➢ Akzentzeichen

➤ für wörtliche Rede im Deutschen: doppelte Anführungszeichen „ – “ (99 – 66)

➤ Bei Belletristik: „Guillemets“ üblich (spitze Klammern)

➤ Zusatzpaket: `csquotes`

```
\usepackage[german]{babel}
\usepackage[german=guillemets]{csquotes}
\defineshorthand{"‘}{\openautoquote}
\defineshorthand{"’}{\closeautoquote}
\begin{document}
"‘Ein Satz mit einem "‘Wort"’"’.
\end{document}
```

Ergebnis

Ligaturen

➤ Zusammenziehung mehrerer Buchstaben → zusätzliches Zeichen in Symboltabelle beim Druck

➤ gebräuchlich: ff, fi, fl (auch ffi, ffl), (auch ß ?)

➤ Beispiel: Pflug anstelle von Pflug

➤ keine Ligatur darf bei Silben- und Wortfugen benutzt werden: `"|`
`Auf"|lage` Auflage (statt: Auflage)
`Schaf"|fell` Schaffell (statt: Schaffell)

Kerning

➤ Zusammenziehen von Buchstaben in Abhängigkeit der konkreten Buchstabenform

➤ Vo – Vo

➤ geeignete Buchstabenkombinationen sind in der Fontdefinition festgelegt, für Nutzer tabu

➤ bei professionellen Zeichensätzen: mehrere Tausend Buchstabenpaare

Silbentrennung

- geht in der Regel automatisch und richtig
- (zusätzliche) Trennungen ermöglichen:
  - `"-`: `Worttrenn"-stelle`
  - `\-`: nur diese Trennstelle im Wort ist erlaubt
  - `\hyphenation{...}`: Ausnahmelexikon, gilt für gesamten Text
    `\hyphenation{Tu-to-ri-um}`
- Trennungen verhindern:
  - `~`: `3~cm`
  - `\mbox{...}`: LR-Box
    `\mbox{(01234) 567 890}`
  - `\hyphenation{...}`: Wörter ohne Trennstellen angeben
    `\hyphenation{Linux}`

Zusatzpaket `microtype`

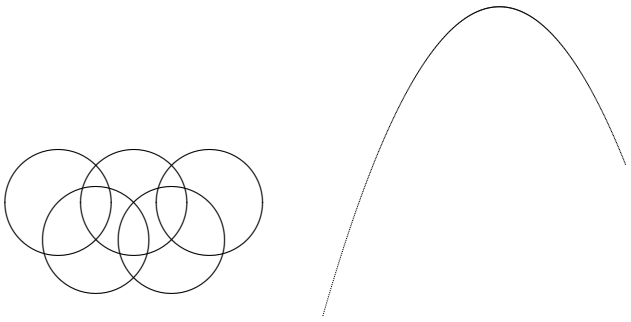
- sehr neu!
- nur mit PDFTEX und LuaTEX
- realisiert automatisch eine Menge von mikrotypografischen Anpassungen (z.B. optischer Randausgleich)
- bietet eine riesige Anzahl von Steuerungsmöglichkeiten per Kommando

7. Grafiken und Bilder

7.1 Generierung von Grafiken: `picture`-Umgebung

```
\begin{picture}
spezielle Befehle
\end{picture}
```

```
\setlength{\unitlength}{1cm}
\begin{picture}(8,4)
\put(1,1){\circle{3}}
\put(2,1){\circle{3}}
\put(0.5,1.5){\circle{3}}
\put(1.5,1.5){\circle{3}}
\put(2.5,1.5){\circle{3}}
\qbezier(4,0)(6,7)(8,2)
\end{picture}
```

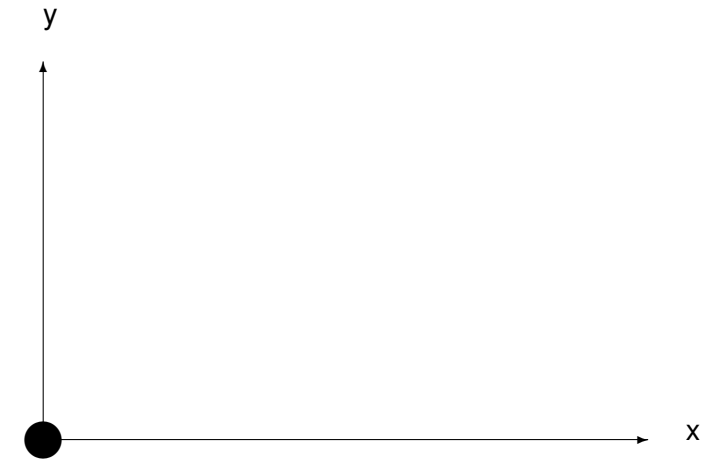


Basis: Koordinatensystem

`\begin{picture}(`*x-einheiten*`,`*y-einheiten*`)`

Vorher: Größe einer Einheit festlegen:

`\setlength{\unitlength}{`*mass*`}`



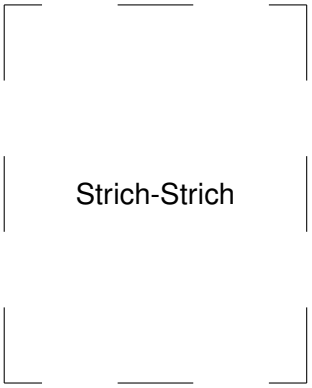
Positionierungsbefehle:

`\put(`*x*`,`*y*`){`*objekt*`}`
`\multiput(`*x*`,`*y*`)(`*x-delta*`,`*y-delta*`){`*anzahl*`}{`*objekt*`}`

Objekte:

- Text `\put(4,5){Text}`
- Rechtecke `\makebox(`*x-einheiten*`,`*y-einheiten*`) [`*position*`]{`*objekt*`}`
  `\framebox(`*x-einheiten*`,`*y-einheiten*`) [`*position*`]{`*objekt*`}`
  `\dashbox(`*x-einheiten*`,`*y-einheiten*`) [`*position*`]{`*objekt*`}`

```
\put(1,1){\framebox(3,2){Mitte}}
\put(0,0){\makebox(0,0)[bl]{Ursprung}}
\put(8,3){\dashbox(4,5){Strich-Strich}}
```



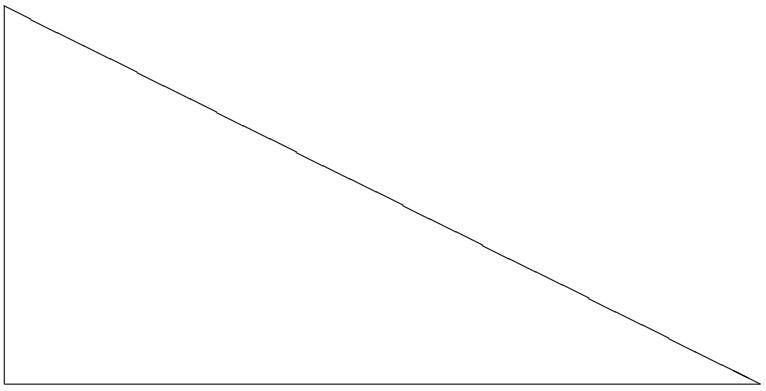
Ursprung

position: `t` top
`b` bottom
`l` left
`r` right

➤ Linien

`\line(`*x-richtung*`,`*y-richtung*`){`*längenprojektion*`}`

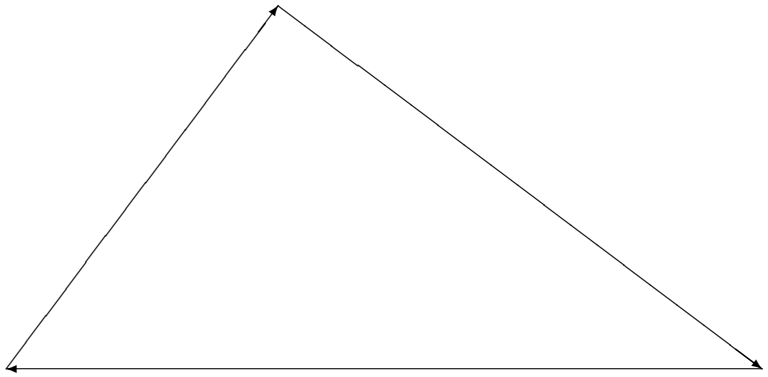
```
\put(0,0){\line(1,0){10}}
\put(0,0){\line(0,1){5}}
\put(0,5){\line(2,-1){10}}
```



➤ Vektoren

\vector(*x-richtung*,*y-richtung*){*längenprojektion*}

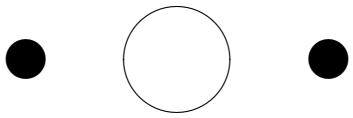
```
\put(10,0){\vector(-1,0){10}}
\put(0,0){\vector(3,4){3.6}}
\put(3.6,4.8){\vector(4,-3){6.4}}
```



➤ Kreise

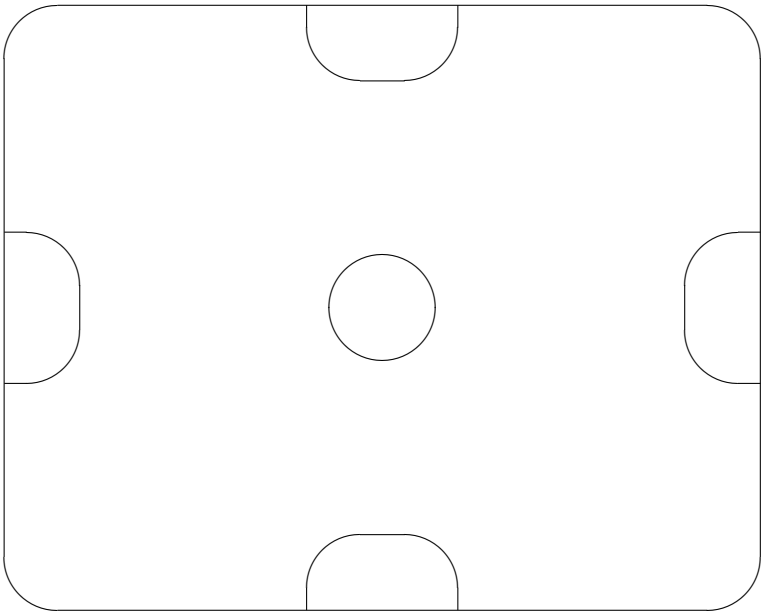
\circle{*durchmesser*}
\circle*{*durchmesser*}

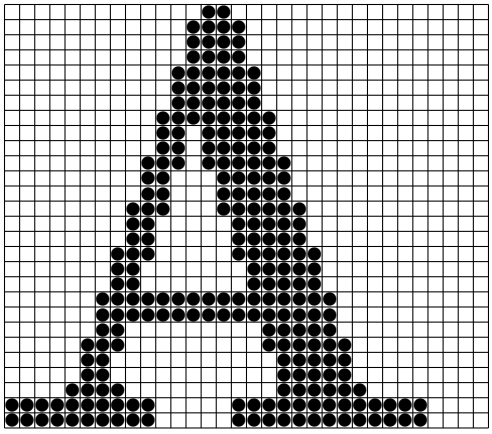
```
\put(3,2){\circle{4}}
\put(1,2){\circle*{2}}
\put(5,2){\circle*{2}}
```

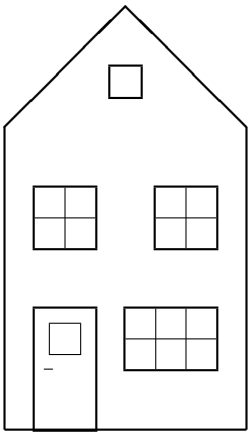


➤ Ovale (Kreisabschnitte)

\oval(*x-einheiten*,*y-einheiten*)[*position*]}







Zusatzpaket `pict2e`

- verbesserte Neudefinition der `picture`-Befehle
- einige interne Beschränkungen fallen weg
- keine neue Nutzerschnittstelle

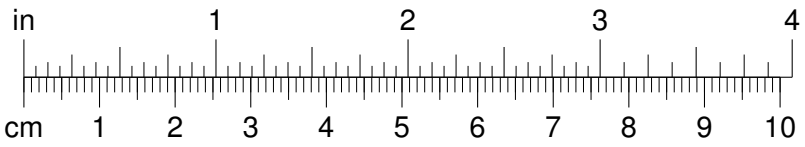
7.2 Generierung von Grafiken: PICTEX

- Zusatzpaket `pictex`
- wesentlich mehr Fähigkeiten als `picture`-Umgebung
- Benutzung wesentlich aufwändiger
- andere, komplexere Befehle mit ausgefallener Syntax

```
\beginpicture
\setcoordinatesystem units <1mm,1mm>
\setplotarea x from 0 to 50, y from 0 to 50
\put {Ursprung} at 0 0
\multiput {.} at 40 20 *10 2 2 *10 1 -3 *10 -3 1 /
\endpicture
```

Ursprung

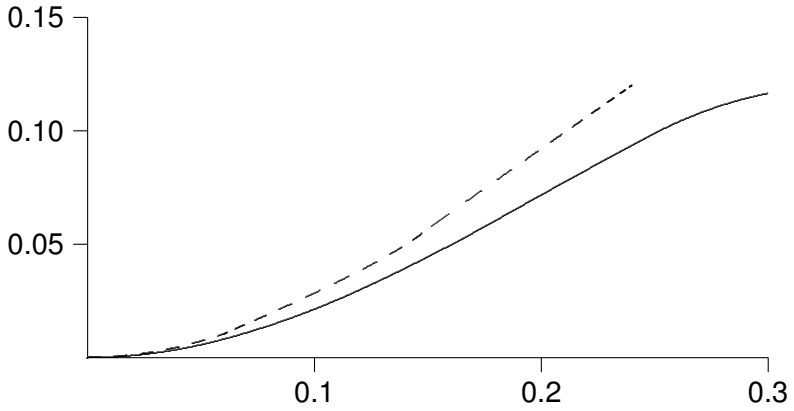
```
\axis bottom ticks width <0.4pt> length <4mm>
 withvalues cm / at 0 / numbered from 1 to 10 by 1
 unlabeled width <0.2pt> length <3mm>
  from 0.5 to 9.5 by 1.0  length <2mm> quantity 101 /
\setcoordinatesystem units <1in,1in>
\setplotarea x from 0 to 4, y from 0 to 0
\axis top ticks width <0.4pt>   length <5mm>
 withvalues in / at 0 / numbered from 1 to 4 by 1
 unlabeled length <4mm> from 0.5 to 3.5 by 1.0
  width <0.2pt> length <3mm> from 0.25 to 3.75 by 0.50
   length <2mm> from 0.125 to 3.875 by 0.250 ...
```




```
\beginpicture
\setquadratic
\setcoordinatesystem units <30cm,30cm>
\setplotarea x from 0.0 to 0.3, y from 0.0 to 0.15
\axis left ticks numbered from 0.05 to 0.15 by 0.05 /
\axis bottom shiftedto y=0 ticks numbered from 0.1
 to 0.3 by 0.1 length <3mm> /
\plot "VERBATIM/beisp32a"
\setdashes
\plot "VERBATIM/beisp32b"
\endpicture
```

beisp32a:

```
0.000000 0.000000 0.030000 0.002109 0.060000
0.008128 0.090000 0.017560 0.120000 0.029857
0.150000 0.044401 0.180000 0.060478 0.210000
0.077226 0.240000 0.093538 0.270000 0.107799
0.300000 0.116453
```



7.3 Weitere Pakete

`epic`, `eepic` Erweiterung der `picture`-Umgebung

`multibox` wiederholte Ausgabe von Objekten

`tree` Zeichnen von (binären und ternären) Bäumen

`bar` Konstruktion von Balkendiagrammen

`cuves` Darstellung von beliebigen Kurven

`pgf` komplexes Paket zur „Programmierung von Grafiken“ (portable graphics format) definiert TEX-Kommandos, die von den Drivern verarbeitet werden

`tikz` LATEX-Frontend für pgf

```
\tikz \draw[thick,rounded corners=8pt]
 (0,0) -- (0,2) -- (1,3.25) -- (2,2) -- (2,0)
 -- (0,2) -- (2,2) -- (0,0) -- (2,0);
```

Ergebnis

Zusatzpaket `pstricks`

- extensive Nutzung von Postscript-Fähigkeiten
- Plotten, Knoten, Bäume, Schaltbilder
- 3D-Darstellungen
- Schattierung und Farbverläufe
- geographische Projektionen
- ...

Beispiele

Resüme:

- ➤ sehr bis extrem aufwändig
- ➤ Ergebnisse unbefriedigend bis ok

7.4 Benutzung fertiger Grafiken und Bilder

- ➤ Erzeugung mit beliebiger Software
- ➤ Format: `eps` bzw. `pdf`/`jpg`/`png`
- ➤ Verarbeitung mit `latex` bzw. `pdflatex`

Zusatzpaket `graphicx`

Kommando

`\includegraphics[`*optionen*`]{`*file*`}`

Optionen: `schlüsselwort=`*wert*

`scale`	Skalierungsfaktor (negative Zahl: Spiegelung)
`height`	Bildhöhe
`width`	Breite
`angle`	Drehwinkel im Uhrzeigersinn
`clip`	abschneiden an Bildkanten
`draft`	Rechteck als Bildersatz
`bb`	Bildmaße (Bounding Box)

wenn andere Bildformate vorliegen: extern konvertieren

oder:

`\DeclareGraphicsExtensions{`*suffix, ...*`}`
`\DeclareGraphicsRule{`*suffix*`}{`*typ*`}{`*groessendatei*`}`
`{`*konvertierungskommando*`}`

groessendatei enthält die Bildmaße

mit geeigneten Kommandos generieren oder per Hand bestimmen

Alternative: `\includegraphics[bb=0 0 2048 1536,width=5cm]{bild}`

Beispiel:

```
\DeclareGraphicsExtensions{.eps,.ps,.eps.gz,.ps.gz,
    .eps.Z,.tif,.fig}
\DeclareGraphicsRule{.eps.gz}{eps}{.eps.bb}
    {‘gunzip -c #1}
\DeclareGraphicsRule{.tif}{eps}{}{‘convert #1 ’eps:-’}
\DeclareGraphicsRule{.fig}{eps}{}{’fig2dev -L ps #1}
```

7.5 Umfließen von Bildern

- Ziel: automatisches „Umfließen“ von Bildern und Tabellen durch den Text eines Absatzes
- `wrapfig`-Paket
- neue Umgebung:

  `\begin{wrapfigure}{`*position*`}{`*breite*`}`
  bild
  `\end{wrapfigure}`

- Position: `l`, `r`, `i` (innen), `o` (außen)

```
\begin{wrapfigure}{l}{3cm}
\includegraphics[width=3cm]{penguin}
\end{wrapfigure}
Die Chemnitzer Linux-Tage sind eine Veranstaltung
für jedermann rund um das Thema Linux und
OpenSource. Sie werden durchgeführt vom
IN-Chemnitz, der CLUG, dem Rechenzentrum
...
```

Ergebnis

Alternatives Paket zum Umfließen: `cutwin`

- neue Umgebung:

  `\begin{cutout}{`*zeilen-vorher*`}{`*rand-links*`}{`*rand-rechts*`}{`*zeilen-fenster*`}`
  ...
  `\end{cutout}`

- `shapedcutout`
  ermöglicht selbstdefinierte Makros für die Ränder anzugeben
  Beispiel
- weitere interessante Pakete: `picins`, `floatflt`

7.6 Farben

Zusatzpaket `color`

Farbmodelle

`named`-Farbmodell

- LaTeX „kennt“ zunächst 8 Farben: `white`, `black`, `red`, `green`, `blue`, `yellow`, `magenta` und `cyan`
- `dvips` kennt weitere 68 Farben (`Goldenrod`, `RoyalBlue`, ...)
- bei `pdflatex` `color`-Paket mit Option `dvipsnames` laden

`rgb`-Modell

- additive Farbmischung, Bildschirmfarben
- jeweilige Anteile an den Grundfarben rot, grün, blau
- Werte zwischen 0 und 1
- (0,0,0) = schwarz, (1,1,1) = weiß

`cmyk`-Modell

- subtraktive Farbmischung, Druckerfarben
- jeweilige Anteile an den Farben zyan, magenta, gelb, schwarz
- Werte zwischen 0 und 1
- (0,1,1,0) = rot

`gray`-Modell

- Graustufen, zwischen 0 und 1
- 1 = weiß, 0 = schwarz

Definition von Farben

`\definecolor{`*name*`}{`*farbmodell*`}{`*werte*`}`

```
\definecolor{gruen}{named}{green}
\definecolor{meinlila}{named}{Plum}
\definecolor{rosa}{cmyk}{0.0,0.5,0.5,0.0}
\definecolor{hellgrau}{gray}{0.95}
```

Benutzung von Farben

\color{*farbe*}
\textcolor{*farbe*}{*text*}
\colorbox{*farbe*}{*text*}
\fcolorbox{*rahmenfarbe*}{*farbe*}{*text*}
\pagecolor{*farbe*}

```
\definecolor{meinlila}{named}{Plum}
\textcolor{green}{\emph{Das ist ein grüner Text.}}
\textcolor{meinlila}{So sieht mein Lila aus.}
\fcolorbox{black}{red}
    {\textcolor{white}{\textbf{ACHTUNG}}}
```

Das ist ein grüner Text. So sieht mein Lila aus.

ACHTUNG

```
\linethickness{6mm}
\color{black}\put(0,12){\line(1,0){36}}
\color{red}\put(0,6){\line(1,0){36}}
\color{yellow}\put(0,0){\line(1,0){36}}
```

Zusatzpaket `xcolor`

- Neuentwicklung
- ermöglicht Definition der Farbsättigung
- Farbverläufe, -abstufungen
- ...

Beispiel

8. Tabellen

8.1 Funktionelle Erweiterungen von `tabular`

bisher bekannt:
Formatierungsangaben `l`, `c`, `r`, `p`, `@{.}`
`multicolumn`, `hline`

Zusatzpaket `array`

weitere Formatierungsangaben

`m`{breite}	Spalte fester Breite (wie `p`), Inhalt vertikal zentriert
`b`{breite}	Spalte fester Breite (wie `p`), Inhalt fußbündig
`>`{dekl}	Deklaration wird direkt vor den Spalteninhalt eingefügt
`<`{dekl}	Deklaration wird direkt nach dem Spalteninhalt eingefügt

```
\begin{tabular}{lp{2cm}m{2cm}b{2cm}}
Ausrichtung & oben & mitte & unten \\ \hline
Mitte &
 oberste Zeile ausgerichtet &
  vertikal zentriert &
   unterste Zeile ausgerichtet \\
\end{tabular}
```

Ausrichtung	oben	mitte	unten
Mitte	oberste Zeile ausgerichtet	vertikal zentriert	unterste Zeile ausgerichtet

```
\begin{tabular}{>{\scriptsize}r >{\bfseries}l l}
Nr. & Name & Vorname \\ \hline
1 & Müller & Max \\
2 & Meier & Michael \\
3 & Mustermann & Otto
\end{tabular}
```

Nr.	**Name**	Vorname
1	**Müller**	Max
2	**Meier**	Michael
3	**Mustermann**	Otto

Zusatzpaket `tabularx`

`X` Breite wird aus der Gesamttabellenbreite automatisch berechnet

`\begin{tabularx}{`gesamtbreite`}{`spaltendekl`}`
...
`\end{tabularx}`

```
\begin{tabularx}{.9\textwidth}{l X}
Name & Aufgabengebiete \\
\hline
Müller & \LaTeX-Programmierung, Gestaltung von
    Webseiten, Durchführung von
    Kursen, Betreuung von Diplomarbeiten\\
Meier & Administrator Supercomputing \\
\end{tabularx}
```

Name	Aufgabengebiete
Müller	LaTeX-Programmierung, Gestaltung von Webseiten, Durchführung von Kursen, Betreuung von Diplomarbeiten
Meier	Administrator Supercomputing

auch mehrere `X` möglich: die entsprechenden Spalten teilen sich *gleichmäßig* den verbleibenden Platz

```
\begin{tabularx}{10cm}{l|X|X|X|c}
Name & T1 & T2 & T3 & Fl \\ \hline
Müller & abcde & 0 & 0 & x \\
Meier  & mn & ppq & xyxyxyxyx & \\
Schulze & s & iillii & 1 & x \\
\end{tabularx}
```

Name	T1	T2	T3	Fl
Müller	abcde	0	0	x
Meier	mn	ppq	xyxyxyxyx	
Schulze	s	iillii	1	x

Zusatzpaket `longtable`

Standardtabellen kennen keinen Seitenumbruch

mit `longtable`: Tabelle wird an Seitengrenzen umgebrochen
Definition von Tabellenkopf und -fuß möglich

`\begin{longtable}{`spaltendekl`}`
kopfzeile(n) `\endhead`
fusszeile(n) `\endfoot`
...
`\end{longtable}`

```
\begin{longtable}{l|l|l|}
\hline Name, Vorname & Titel & Fakultät,
   Professur\\ \hline\hline \endhead
\hline \endfoot
Hoffmann, Karl Heinz  & Prof. Dr.  &
    NW, Institut für Physik \\
Woschni & Prof. Dr. Dr. & ET/IT \\
Kreissig, R. & Prof. Dr. & MB, Festkörpermech. \\
\end{longtable}
```

Ergebnis

weitere Zusatzpakete für Tabellensatz:

`supertabular` ähnlich wie `longtable`, aber Spaltenbreiten werden auf jeder Seite neu berechnet

`ltxtable` vereinigt die Funktionalität von `longtable` und `tabularx`

`dcolumn` Ausrichten von Spalten am Dezimalpunkt (oder anderem Zeichen)

`hhline` Erzeugen von horizontalen und vertikalen Linien mit „Eckenbildung“

`multirow` mehrzeilige Spalten (analog zu `multicolumn`)

`booktabs` „saubere“ waagerechte Linien

8.2 Farbige Tabellen

Einzelne Zellen einfärben: siehe Kapitel 6

Farbige Spalten und Zeilen:
Zusatzpaket `colortbl`

`\columncolor[`*farbmodell*`]{`*farbe*`}`
`\rowcolor[`*farbmodell*`]{`*farbe*`}`

Kommando `columncolor` nur als Spaltenspezifikation > (`array`-Paket) benutzen!

```
\begin{tabular}{|r|>{\columncolor{red}}l|r|}
1 & Müller & 12345 \\
2 & Schulze & 9988 \\
3 & Maier & 815
\end{tabular}
```

```
\begin{tabular}{r >{\columncolor{red}}lr}
1 & Müller & 12345 \\ 2 & Schulze & 9988 \\
3 & Maier & 815 \\ \hline\hline
\rowcolor{yellow} & \emph{Summe:} & 23148
```

9. Zeichensätze

9.1 Allgemeines

- Jahrhundertelange Geschichte
- viele Fachausdrücke
- künstlerische und physiologisch-psychologische Aspekte
- DIN 16518: 11 Klassen
- TEX-Schriften nach amerikanischen Gepflogenheiten und Neuentwicklung von KNUTH
- Klassifizierungen:

  Brotschriften — Akzidenzschriften (Zierschriften)
  Proportionalschriften — Fixschriften
  Serifenschriften — Sans Serifenschriften
  Senkrechte Schriften — Kursive Schriften
  Normalschriften — Fettschriften
  Alphabetzeichensätze — Symbolzeichensätze

Übersicht der Zeichensatzeinteilung bei TEX-Fonts – bezogen auf die (veraltete) Familie „Computer Modern“ (`cm`) von Knuth

9.2 Schriftfamilien

Anderer Ansatz: Einteilung der Schriften in Familien

- Mitglieder einer Familie unterliegen demselben Designprinzip
- unterscheiden sich in 5 Parametern:
- Schriftform (=Schriftlage), „font shape“
  - aufrecht
  - kursiv (italic) / geneigt (slanted, oblique)
  - Kapitälchen
  - Outline
  - schattiert
- Schriftstärke (=Strichstärke), „font series“
  - medium, fett
  - ultraleicht, extraleicht, . . . ultrafett

- Schriftbreite
  - schmal, normal, weit
- Schriftgrade (=Schriftgröße), „font sizes“
  - . . . , 11 Punkte, . . .
- Kodierschema
  - Größe des Zeichensatzes (der Zeichensatztabelle): 128, 256
  - Anordnung der Zeichen in der Tabelle
- in der Regel enthält eine Familie nur einige Ausprägungen der Möglichkeiten

Jedes dargestellte Zeichen ist aus einer konkreten Zeichensatzdatei, die einer bestimmten *Schriftform*, *Schriftstärke*, *Schriftbreite*, einem *Schriftgrad* und einem *Kodierschema* aus einer bestimmten Schriftfamilie entspricht.

LaTeX hat standardmäßig 3 *Schriftfamilien*:

- Serifenschrift für Texte – `\textrm` bzw. `\rmfamily`
- Serifenlose Schrift – `\textsf` bzw. `\sffamily`
- Schreibmaschinenschrift – `\texttt` bzw. `\ttfamily`

welche Schriften das wirklich sind, hängt von Dokumentklasse, gewählten Zusatzpaketen oder anderen Befehlen ab

Voreinstellung: „Computer Modern Roman“, „Computer Modern Sans“, „Computer Modern Typewrite“

die 3 benutzten Familien sollten immer harmonisch zusammenpassen!

Schriftformen in LaTeX: `\textup`/`\upshape` (ungebräuchlich, weil Standard), `\textit`/`\itshape`, `\textsl`/`\slshape`, `\textsc`/`\scshape`

Schriftserien in LaTeX: `\textmd`/`\mdseries`, `\textbf`/`\bfseries` (betr. Schriftbreite und -stärke zusammen)

Schriftgrade in LaTeX: `\tiny` ... `\Huge` (relativ), `10pt`, `11pt`, `12pt` (Grundeinstellung absolut)

Kodierschema in LaTeX: `OT1` (Computer Modern), `T1` (ec-Schriften), . . .

9.3 Änderung der voreingestellten Textfonts

interne Variable, die das Verhalten von Fontbefehlen beeinflussen:

Variable	*Voreinstellung*	*Beschreibung*
`\encodingdefault`	`OT1`	Kodierschema Basisschrift

`\familydefault`	`\rmdefault`	Schriftfamilie Basisschrift
`\seriesdefault`	`m`	Schriftserie der Basisschrift
`\shapedefault`	`n`	Schriftform der Basisschrift
`\rmdefault`	`cmr`	Schriftfamilie, ausgewählt von `\rmfamily` und `\textrm`
`\sfdefault`	`cmss`	Schriftfamilie, ausgewählt von `\sffamily` und `\textsf`
`\ttdefault`	`cmtt`	Schriftfamilie, ausgewählt von `\ttfamily` und `\texttt`
`\bfdefault`	`bx`	Schriftserie, ausgewählt von `\bfseries` und `\textbf`
`\mddefault`	`m`	Schriftserie, ausgewählt von `\mdseries` und `\textmd`
`\itdefault`	`it`	Schriftform, ausgewählt von `\itshape` und `\textit`
`\sldefault`	`sl`	Schriftform, ausgewählt von `\slshape` und `\textsl`
`\scdefault`	`sc`	Schriftform, ausgewählt von `\scshape` und `\textsc`
`\updefault`	`n`	Schriftform, ausgewählt von `\upshape` und `\textup`

Änderung:

`\renewcommand{\`*variable*`}{`*wert*`}`

```
\renewcommand{\familydefault}{cmss}
```

setzt den gesamten Text in Computer Modern Sans Serif

für Nutzer ist das alles ziemlich aufwändig, deshalb fertige Zusatzpakete benutzen.

- Concrete-Schriftfamilie
  Entwicklung von Knuth, passend zur Mathematikschrift `euler` von Zapf
- Pandora-Schriftfamilien
  Entwicklung von Billawala: Pandora Roman, Pandora Sans

  ```
  \usepackage{pandora}
  ```

  . . . fertig!

  `pandora.sty`:

```
...
\renewcommand{\encodingdefault}{OT1}
\DeclareFontSubstitution{OT1}{panr}{m}{n}
\renewcommand{\rmdefault}{panr}
\renewcommand{\sfdefault}{pss}
\renewcommand{\bfdefault}{b}
\renewcommand{\itdefault}{sl}
...
```

Und so siehts aus

➤ Altdeutsche Schriften: Zusatzpaket `oldgerm`
Entwicklung von Haralambous
enthält „Gotisch“ (Textur), „Schwabacher“, „Fraktur“ und Initialen

```
{\gothfamily Das ist gothisch.}
{\frakfamily Das ist Fraktur.}
{\swabfamily Das ist Schwabacher.}
```

Ergebnis ... und die ganze Schönheit

9.4 Postscript-Fonts

aus Nutzersicht: genauso Zusatzpakete mit verfügbaren Schrift(familien)

Paket	Serifenlose Schrift	Serifenschrift	Typewriterschrift
`times`	Helvetica	Times	Courier
`palatino`	Helvetica	Palatino	Courier
`helvet`	Helvetica		
`avant`	AvantGarde		
`newcent`	AvantGarde	NewCentury-Schoolbook	Courier
`bookman`	AvantGarde	Bookman	Courier

`helvet.sty` enthält (als Kern): `\renewcommand{\sfdefault}{phv}`

d.h. die eingestellte Serifen- und die Schreibmaschinenschrift werden nicht geändert!

Wenn man *alles* serifenlos setzen will: `\renewcommand{\familydefault}{phv}`

oder noch kürzer: `\sffamily`

9.5 Physische Fontfiles

- welche Dateien des Fontsystems werden beim Formatieren und Betrachten benutzt?
- alle Fontdateien (der TEX-Installation) stehen in Verzeichnissen unter `/usr/share/texmf/fonts`
- zusätzlich werden noch `/usr/local/share/texmf/fonts` und `~/texmf/fonts` verwendet
  (s. Kapitel TEX-nisches)
- Formatierer (`latex` bzw. `pdflatex`) benötigen
  - `fd`-Files: „Font Definition“ (ASCII): Abbildung der gewünschten Zeichencharakteristika auf konkrete Filenamen
    z.B. `/usr/share/texmf/tex/latex/base/ot1cmr.fd`
  - `tfm`-Files: „TeX Font Metric“ (binär)
    für jedes Zeichen: Breite und Höhe der Zeichenbox, ggf. Kerninginformationen, nicht: Zeichenform!
    z.B. `/usr/share/texmf/fonts/tfm/public/cm/cmr10.tfm`
- DVI-Konverter, -Viewer und `pdflatex` benötigen konkrete Zeichenformen, bezogen auf die Auflösung des Ausgabegeräts
  - große Mengen von `pk`-Files vorhalten
  - dynamisch berechnen (`maketexpk`, METAFONT)
  - bei aktuellen TEX-Systemen stehen auch die „Computer Modern“ und die ec-Fonts als Postscript-Font bereit
  - `pfb`-Files: kleine Postscriptprogramme zur Zeichenerzeugung, teilweise auch `vf`-Files

  $\Rightarrow$ das erzeugte PS-File (oder PDF) enthält die Fonts – kann zu lizenzrechtlichen Problemen führen

9.6 Integration und Benutzung zusätzlicher Fonts

- „open source“ Schriften
- kommerzielle Schriften (CD, . . .)
- kostenfrei nutzbare, z.B. von `www.freefont.de`

1. Fall: freie Schrift
z.B. ftp-Server/CTAN: `/pub/tex/fonts/augie`

1. die angebotenen Files kopieren
   - `tex/*` ⇒ `~/texmf/tex/latex/augie`
   - `tfm/*` ⇒ `~/texmf/fonts/tfm/augie`
   - `dvips/*` ⇒ `~/texmf/dvips/config`
   - `type1/*` ⇒ `~/texmf/fonts/type1/augie`
   - `vf/*` ⇒ `~/texmf/fonts/vf/augie`
2. ein Anwendungsbeispiel `sample.tex` ausdenken
   dabei die Hinweise in `doc/*` beachten

```
\documentclass{scrartcl}
...
\begin{document}
\fontfamily{augie}\selectfont
...
```

3. formatieren: `latex sample`
4. `dvips -o sample.ps -u +augie sample`
5. fertig!

2. Fall: Kommerzielle Schrift von CD
und
3. Fall: Freefont-Schrift
Fa. SoftMaker: bietet „kostenlose Schrift des Monats“: `http://www.freefont.de/`

Schritte:

1. Download → Lizenzbestimmungen akzeptieren → Format auswählen: Postscript Type1-Format für PC (Truetype könnte man prinzipiell komvertieren, ist aber schwierig!)
   → `file.zip`
2. Installationsanleitung
   - siehe DANTE-FAQ
3. kein Automatismus, sondern viel Know-How (und Intuition) notwendig
4. Ergebnis: Testseite mit Helium-Font

9.7 Zusatzpakete für Zeichensätze

9.7.1 `eurosym`-Paket

- Paket enthält Euro-Zeichen
- passt sich der aktuellen Schriftgröße an
- auch fett und kursiv
- `\euro`
- einige andere Pakete mit Euro-Zeichen

`10 \euro` 10 €

9.7.2 `textcomp`-Paket

- Sammlung von Sonderzeichen teilweise mit syntaktischer Bedeutung, Ziel ist immer die Ausgabe des Zeichens selbst
- jedes Zeichen hat einen Namen
- Beispiel: `\textdollar` $

Übersicht

9.7.3 `pifont`-Paket

- Zeichensammlung „Zapf Dingbats“
- neuer Befehl: `\ding{`*zahl*`}`
- Beispiel: `\ding{37}` ☎
- `\dingline{`*zahl*`}`, `\dingfill{`*zahl*`}`
- `dinglist`-Listenumgebung

Übersicht

10. Hypertext

- ➤ Technik zur nicht-sequentiellen Informationsgewinnung
- ➤ Widerspruch zum Ziel „druckbare Dokumente“ (Seitenorientierung)
- ➤ Ansätze für Nicht-Linearität schon lange üblich:
  Inhaltsverzeichnis, Index, Lexikon, ... $\Rightarrow$ Druck
- ➤ Hypertext im engeren Sinn: WWW + Internet + HTML
- ➤ PDF kennt auch „Links“

1. Lösungsidee: Konvertierung LaTeX $\rightarrow$ HTML?

Was wäre dabei zu tun?

- ➤ Textformatierung (Setzen von Absätzen, Ausrichtungen, Listen, Tabellen, ...)
- ➤ Behandlung von Sonderzeichen
- ➤ Verarbeitung von Formeln, Bildern und Grafiken (`.eps` $\rightarrow$?)
- ➤ Erzeugung von verlinkten Strukturen: Inhaltsverzeichnis als Sammlung von Links, ebenso Verweise im Text
- ➤ ...

Es gibt solche Wunderwerkzeuge:

1. `latex2html`

   Implementierung der Funktionalität der Standardklassen und einiger Zusatzpakete in Perl
   alle in HTML unbekannten Objekte werden als Bild erzeugt

   Beispiel: PDF-File

   ```
   latex2html sammlung.tex
   ```

   Ergebnis der Konvertierung

   veraltet!

2. TeX4HT
   - ➤ „highly configurable TeX-based authoring system for producing hypertext“
   - ➤ außer HTML/XHTML auch andere Ausgabeformate: OpenOffice, DocBook, TEI, JavaHelp
3. HeVeA: `hevea.inria.fr`
   - ➤ geschrieben in Objective Caml
   - ➤ für Linux + Windows

2. Lösungsidee: PDF benutzen

- `pdflatex`
- Besonderheit bei Bildern: kein EPS, sondern nur PDF, PNG, JPG
- Fontauswahl
  - Bitmap-Fonts (=Standard in TEX: `cm`) vermeiden (Qualitätsproblem)!
  - unbedingt T1-Fonts benutzen!
    `\usepackage[T1]{fontenc}`
    * Postscript-Fonts
    * oder: `cm`-Fonts im T1-Format: `cm-super`
    * oder: zusätzlich `ae`-Paket (Anpassung von `cm` an T1) `\usepackage{ae}`
- bei Konvertierung `ps` → `pdf`:
  `dvips -Ppdf -G0`

Erzeugung von Links

Zusatzpaket: `hyperref`

Wirkung:
1. automatisch
   - Umwandlung sämtlicher Querverweise in Hyperlinks
   - Umwandlung von Verweisen auf Fussnoten, Literaturverzeichnis etc. in Hyperlinks
   - Verlinkung von Verzeichnissen und Index
2. explizit
   - Einfügen von Hyperlinks:
     `\href{URL}{text}`
     URL kann sein:
     - Link auf anderes PDF-File:
       ```
       \href{/usr/local/.../beameruserguide.pdf}%
       {Dokumentation Dokumentklasse beamer}
       ```
       Dokumentation Dokumentklasse beamer
     - Link auf Shellskript, z.B. xterm starten
       ```
       \href{run:script/xterm.sh}%
       {Start eines xterm-Fensters}
       ```
       Start eines xterm-Fensters
       Sicherheitsproblem!!!
   - Definition interner Links
   - Programmieren von PDF-Formulare
   - Konfiguration der Acrobat-Menüs

- ...

Optionen des `hyperref`-Pakets:

- `colorlinks`: Links farbig (statt durch Rahmen gekennzeichnet)
- `urlcolor=red`: Festlegen der Linkfarbe
- `bookmarks`: Erzeugung von Bookmarks
- `bookmarksopen`: „Unterverzeichnisse“ sichtbar
- `pdfpagemode=FullScreen`: Start des Viewers
- ...

Quelltext dieses Skripts:

```
\usepackage[%
bookmarksopen,%
colorlinks,%
urlcolor=green,%
pdfpagemode=none%
]{hyperref}
```

```
...
\hypersetup{pdfsubject={LaTeX für Fortgeschrittene}}
\hypersetup{pdfauthor={Wolfgang Riedel}}
\hypersetup{pdftitle={Kurs}}
```

Ergebnis von `pdflatex ...`:

- PDF-File mit entsprechendem Inhalt
- Dokumentinfos
  (`File` → `Document Proporties` → `Summary`)
- Bookmarks (= aktives Inhaltsverzeichnis)
- Thumbnails (= optisches Seitenverzeichnis)
- der PDF-Browser muss das aber können ...

Weitere Fähigkeiten des `hyperref`-Pakets:

- erweiterte Gliederungskommandos
  `\section`[toc/head][bookmark]{text}
  (Zusatzpaket `hypmbsec`)
  Beispiel: `\section[\LaTeX][LaTeX]{\LaTeX}`
- Erzeugung spezieller Thumbnails
  (Zusatzpaket `thumbpdf`)
- ...

11. Präsentationen

11.1 Vorbemerkungen

- optisch wirksamer Teil eines Vortrags („Folien“)
- Form der Präsentation mindestens genauso wichtig wie Inhalt
- Präsentation $\neq$ Lesetext
  - Gliederung der Informationen in übersichtliche Einheiten
  - Stichworte, keine vollständigen Sätze
  - ...

$\Rightarrow$ Anforderungen an das Layout

- gut lesbarer Font $\Rightarrow$ serifenlos
- „große“ Schrift
- Querformat (Zeilenlänge!)
- feste Seitengröße

Weiter zu beachten:

- grafische Effekte (Einblendung, Farben, Animation ...)
  sparsam einsetzen (nur bei methodischer Notwendigkeit)
- Corporate Identity (Wiedererkennungswert)
- Navigationsmöglichkeiten einplanen (über explizite Buttons oder/und Hyperlinks, evtl. auch über Inhaltsverzeichnis)
- Erste Folie(n): Titel, Autor, Problem- bzw. Zielstellung, ...
- Letzte Folie: „Danke für die Aufmerksamkeit“ o.ä.
- allgemeine Typografieregeln einhalten (z.B. möglichst wenig unterschiedliche Fonts)
- Folienmenge entsprechend Zeitrahmen
- Fortschrittsanzeige (Foliennummer, Balken, aktives Inhaltsverzeichnis, ...)
- Vollbildmodus

11.2 PDF-Ausgabe

- ➤ optisch attraktives, plattformunabhängiges Ausgabeformat
- ➤ Vollbildanzeige
- ➤ Hyperlinks (Zusatzpaket `hyperref`)

einfache Erzeugung dank `pdflatex`

was fehlt:

- ➤ variable Seitenübergänge (Einblenden etc.)

11.3 Herkömmliche Dokumentklassen

`slides`
`seminar`
`foils`

- ➤ Font-, Formateinstellungen okay
- ➤ keine Effekte
- ➤ → veraltet

11.4 Dokumentklasse `texpower`

basiert auf `seminar`-Klasse

Funktionalität des Pakets:

- ➤ schrittweises Aufdecken einer Seite (sequentiell): `\pause`
- ➤ schrittweises Aufdecken innerhalb eines Absatzes: `\stepwise`{ `\step`{text1} `\step`{text2} ... }
- ➤ schrittweises Aufdecken innerhalb einer Grafik, einer Tabelle oder Formel
- ➤ Aufdecken in beliebiger Reihenfolge
- ➤ „aktueller“ Text in anderer Farbe, Textstücke unsichtbar machen
  Demo des Autors
- ➤ verschiedene Seitenübergänge
  `\pageTransitionBoxI`, `\pageTransitionSplitHO` ...
  auch: Vertikal, Wipe, Glitter ...
  (PDF-Viewer muss das unterstützen! `xpdf`?)

- festgelegte Farbmodi für Folienelemente (Normaltext, Hervorhebung, Formel, ...), normalfarbig, blass, ...

11.5 Dokumentklasse `beamer`

Besondere Fähigkeiten:

- Auswahl eines vorgefertigten Layouts (Seitenaufteilung, Farben, Fonts, ...): „Thema“
- Navigationsleiste automatisch generiert
- Anzeige des Inhaltsverzeichnis bei jedem neuen Kapitelanfang, Hervorhebung des aktuellen Kapitels
- Aufzählungen können schrittweise eingeblendet werden
- Erzeugung einer Druckversion aus gleicher Quelle, dabei „Fallunterscheidung“: `\mode<...>`
- alles außerhalb `frame`-Umgebung bzw. `frame`-Kommando landet nur in Druckversion
- ...

Ein erster Versuch:

```
\documentclass{beamer}
\usepackage{ngerman}
\title{Mein Beispiel}
\author{ich}
\begin{document}
\begin{frame}
\maketitle
```

```
\end{frame}
\begin{frame}
\tableofcontents
\end{frame}
\begin{frame}
\section{Erstes Kapitel}
\frametitle{Inhalt des ersten Kapitels}
Das ist eine Seite
\end{frame}
```

```
\begin{frame}
\section{Zweites Kapitel}
Das ist eine neue Seite
\end{frame}
\end{document}
```

Ergebnis

Festlegung des Layout über „Thema“:
`\usetheme{`*thema*`}`

z.B. „Berlin“

Demo verschiedener Themes

Vorlagen für Beamer-Anwendung: http://www.tu-chemnitz.de/urz/anwendungen/tex/template.html

Layout des Templates von C. Schubert

12. System-TEX-nisches

12.1 TEX-Implementationen

- Distribution für alle Plattformen: TeXLive
- *(veraltet:)* Linux/UNIX: teTeX, ...
- *(veraltet:)* Windows: fpTeX, MikTeX, ...
- ...

12.2 Dateibäume

in der Regel drei Ebenen:

- `/usr/share/texmf/`
  `C:\Programme\TeXLive2007\texmf`

- `/usr/local/share/texmf/`
  `C:\Programme\texmf-local`
- `~/texmf/`
  `U:\texmf`

Untergliederung in allen 3 Bäumen identisch
Dateistrukturen für Pakete, Fonts, Konfigurationsdaten, ...

12.3 Konfiguration

Zentrales Konfigfile: `/usr/share/texmf/web2c/texmf.cnf`
bzw. `C:\Programme\TeXLive2007\texmf\web2c\texmf.cnf`

Inhalt

12.4 Filesuche

bei Formatierung: über `ls-R`-Files

nützliches Werkzeug: `kpsewhich` *`file`*

13. Schnittstellen zu anderer Software

13.1 MS Office

- Kommerzielle Lösung „Word2TeX“ und „TeX2Word“
- Firma Chikrii Softlab
- physisch: Word-AddIns
- Preise ab 50 €

13.2 OpenOffice.org

- 2 Makropakete („Extensions“) für OOo 3

- „Writer2LaTeX“: `http://extensions.services.openoffice.org/project/writer2latex`
- „Calc2LaTeX“: `http://extensions.services.openoffice.org/project/calc2latex`
- OOopict.pl: Konvertierung von Postscript-Ausgaben von OpenOffice.org Draw in die LaTeX-picture-Umgebung (basierend auf Bezier-Kurven)

13.3 Weitere

`http://projekte.dante.de/DanteFAQ/DokumentKonverter`

14. Was fehlt noch?

- Briefe, Serienbriefe
- Programmierung in TeX: Datentypen, Variablen, Ausdrücke, Funktionen, Makros, Umgebungen, ...
- Zusatzpakete, die auf Programmierung basieren
  - `ifthen`: Schleifen, bedingte Formatierung
  - `calc`: Rechenoperationen (Beispiel, Quelltext),
  - ...
- „Richtige“ Mathematik: AMSTeX
- Erweiterungen, Erweiterungen, ...
- Neue TeX-basierte Systeme: ConTeXt, eTeX, XeTeX, Omega, ...

Danke für die Aufmerksamkeit

... und viel Erfolg bei der Anwendung des Gelernten