

L^AT_EX advanced

TikZ, etc.

18. April 2023

Danke Henning (8pridoeh) dass wir deine Folien aus dem
WS14/15 benutzen dürfen :D

Und auch Danke an alle, die zu den Folien und zum Vortrag
beigetragen haben:

- Walter Stieben 4stieben@inf
- Ruben Felgenhauer 4felgenh@inf
- Malte Hamann 1hamann@inf
- Hauke Stieler 4stieler@inf

- 1 Verweise
- 2 Quellcode
- 3 L^AT_EX erweitern
- 4 Math advanced
- 5 TikZ
- 6 VCS & Co.

Verweise

Referenzieren

Referenzieren (Figures)

L^AT_EX-Code:

```
\begin{figure}[t]
  \includegraphics[width=7cm]{images/lichtstrahl}
  \caption{Brechung eines Lichtstrahls.}
  \label{fig:lichtbrechung} % NACH der caption!
\end{figure}
```

% Irgendwo anders

Der Lichtstrahl wird gebrochen, wie
Abbildung `\ref{fig:lichtbrechung}` zeigt.

Ergebnis:

Der Lichtstrahl wird gebrochen, wie Abbildung 3 zeigt.

Richtig Zitieren

BibT_EX & BibL^AT_EX

- Man verwaltet eine BibT_EX-Datei (*.bib) mit Literaturangaben
- Mit `\cite[Seite X]{Referenz}` referenziert man eine solche Angabe, mit optionaler Seitenangabe.
- Mit `\footcite{Referenz}` kann die Referenz direkt in die Fußnote mit hochgestelltem Index geschrieben werden.
- Vor pdf_latex wirft man bib_tex an

Richtig Zitieren

BibL^AT_EX und Biber

Biber:

- Moderner Ersatz für BibT_EX
- Guter UTF-8-Support
- Bessere Verwaltung von Styles
- Mehr Kontrolle über Sortierung
- Nicht überall verbreitet

BibL^AT_EX

- Package zum Einstellen vieler Dinge mittels L^AT_EX-Code
- Einfaches wissenschaftliches Zitieren
- Für den Einsatz mit Biber entwickelt (UTF-8-Support)
- Funktioniert mittelmäßig mit BibT_EX

Richtig Zitieren

BibT_EX

L^AT_EX-Code:

% Im Header

```
\usepackage[backend=bibtex8,style=alphabetic]{biblatex}  
\addbibresource{../literatur}
```

% Beim Zitat

Für die Lösung des Travelling-Salesman-Problems
wurde ein heuristischer Algorithmus `\cite{lin1973}`
gewählt.

% Literaturverzeichnis anzeigen

```
\printbibliography  
\bibliography{literatur}
```

Richtig Zitieren

BibL^AT_EX-Eintrag

BibL^AT_EX-Eintrag: (aus “literatur.bib”)

```
@article{lin1973,  
  author = {Shen Lin and Brian W. Kernighan},  
  title  = {An Effective Heuristic Algorithm for the  
           Travelling-Salesman Problem},  
  journal = {Operations Research},  
  volume = {21},  
  year   = {1973},  
  pages  = {498--516},  
}
```

Richtig Zitieren

BibT_EX-Ergebnis

Ergebnis:

$$\boxed{v_2}$$

v_1

Für die Lösung des Travelling-Salesman-Problems wurde ein heuristischer Algorithmus [LK73] gewählt.

References

- [LK73] Shen Lin and Brian W. Kernighan. “An Effective Heuristic Algorithm for the Travelling-Salesman Problem”. In: *Operations Research* 21 (1973), pp. 498–516.

Quellcode

Code-Highlighten

Mit minted

Ergebnis:

```
class MeineKlasse{  
    private int meineVariable; // Deklaration  
  
    public void meineMethode(){  
        meineVariable = 42; // Initialisierung  
    }  
}
```

Code-Highlighten

Mit lstlisting

L^AT_EX-Code:

```
\usepackage{listings}
\lstset{...} % Style-Einstellungen
% ...
\begin{lstlisting}[caption=Deklaration und Initialisierung von V
class MeineKlasse{
    private int meineVariable; // Deklaration

    public void meineMethode(){
        meineVariable = 42; // Initialisierung
    }
}
\end{lstlisting}
```

Code-Highlighten

Mit lstlisting

Ergebnis:

```
1 class MeineKlasse{
2     private int meineVariable; //
        Deklaration
3
4     public void meineMethode(){
5         meineVariable = 42; //
            Initialisierung
6     }
7 }
```

Listing 1: Deklaration und Initialisierung von Variablen

Code-Highlighten

Mit verbatim

L^AT_EX-Code:

```
\begin{verbatim}
~$ cat .ssh/config
Host fbi
  User 7nachnam
  ForwardX11 yes
  HostName rzssh1.informatik.uni-hamburg.de
  DynamicForward 7777
  #LocalForward 6631 linuxprint.informatik.uni-hamburg.de:631
\end{verbatim}
```

Code-Highlighten

Mit verbatim

Ergebnis:

```
~$ cat .ssh/config
Host fbi
  User 7nachnam
  ForwardX11 yes
  HostName rzssh1.informatik.uni-hamburg.de
  DynamicForward 7777
  #LocalForward 6631 linuxprint.informatik.uni-hamburg.de:631
```

Code-Highlighten

Mit algorithmic (Pseudocode)

L^AT_EX-Code:

```
\begin{algorithmic}
  \IF{some condition is true}
    \STATE do some processing
  \ELSIF{some other condition is true}
    \STATE do some different processing
  \ENDIF
\end{algorithmic}
```

Ergebnis:

```
if some condition is true then
  do some processing
else if some other condition is true then
  do some different processing
end if
```

\LaTeX erweitern

Makros & Umgebungen

Eigene Befehle

L^AT_EX-Code:

```
% TeX-style
\def\heute{Heute ist der \today.}
% LaTeX-style (besser)
\newcommand{\heute}{Heute ist der \today.}
\newcommand{\TikZ}{Ti\textit{k}Z}
% Verwendung
\heute
\TikZ
```

Ergebnis: Heute ist der 18. April 2023.
TikZ

Makros & Umgebungen

Eigene Befehle – Parameter

L^AT_EX-Code:

```
% LaTeX-style
```

```
\newcommand{\bus}[4]{Ein Bus der Linie #1 Richtung  
#2 fährt von der Haltestelle  
#3 um #4 ab.}
```

```
% Verwendung
```

```
\bus{181}{Sternschanze}{Informatikum}{13:37}
```

Ergebnis: Ein Bus der Linie 181 Richtung Sternschanze fährt von Informatikum um 13:37 ab.

Makros & Umgebungen

Eigene Befehle – Optionale Parameter

L^AT_EX-Code:

```
% LaTeX-style
\newcommand{\bus}[2][181]{Linie #1 Richtung #2.}
% Verwendung
\bus{Sternschanze}
\bus[22]{Blankenese}
```

Ergebnis:

Linie 181 Richtung Sternschanze. Linie 22 Richtung Blankenese.

Makros & Umgebungen

Eigene Umgebungen

L^AT_EX-Code:

```
\newenvironment{textttit}
    {\begingroup\ttfamily\itshape}
    {\endgroup}
```

% Verwendung

```
\begin{textttit}
    Dies ist ein Test
\end{textttit}
```

Ergebnis: *Dies ist ein Test*

Packages & Klassen

Eigene Packages

L^AT_EX-Code (meinpackage.sty):

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{meinpackage}[2000/01/01 Mein Package]

\newcommand{\helloworld}{Hello World!}

\DeclareOption{german}{
  \renewcommand{\helloworld}{Hallo Welt!}
}

\ProcessOptions\relax

\endinput
```

Packages & Klassen

Eigene Dokumentenklassen

L^AT_EX-Code (meineklasse.cls):

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesClass{meineklasse}[2000/01/01 Meine Klasse]

\RequirePackage[utf8]{inputenc}
\RequirePackage[T1]{fontenc}
\RequirePackage[ngerman]{babel}
\RequirePackage{lmodern}

\DeclareOption{sans}{
  \renewcommand*\familydefault{\sfdefault}
}

\ProcessOptions\relax

\LoadClass[a4paper,10pt]{scrartcl}

\endinput
```

Packages & Klassen

Benutzung

L^AT_EX-Code:

```
\documentclass[sans]{meineklasse}
```

```
\usepackage[german]{meinpackage}
```

```
\begin{document}  
  \helloworld  
\end{document}
```

Ergebnis:

Hallo Welt!

Mehr Infos: <http://ctan.mirrors.hoobly.com/macros/latex/doc/clsguide.pdf>
https://en.wikibooks.org/wiki/LaTeX/Creating_Packages

Packages & Klassen

Alternative: Include-Dateien

- `\input{<datei>}` lädt den Inhalt von <datei> stumpf, als ob er so im Dokument stehen würde.
- `\include{<datei>}` macht eine neue Seite auf und kann nicht geschachtelt werden, legt aber eigene aux-Dateien an und kann somit Seitenzahlen und Querverweise beachten.

L^AT_EX-Code (beispiel.inc):

Hallo \LaTeX!

L^AT_EX-Code (Hauptdokument):

`\input{beispiel.inc}`

Ergebnis:

Hallo L^AT_EX!

Math advanced

Math advanced

Mehr tolle Mathe-Tricks

L^AT_EX-Code:

Große Klammern gehen auch: `\left(\frac{n^2 + 1}{3} \right)^2` `\`

Aus `\int_a^b \sum_a^b` wird `\displaystyle \int\limits_a^b \sum_a^b`

Ergebnis:

Große Klammern gehen auch: $\left(\frac{n^2 + 1}{3}\right)^2$

Aus $\int_a^b \sum_a^b$ wird $\int_a^b \sum_a^b$

Anmerkung: Über `\everymath{\displaystyle}` kann jeder Mathe-Modus automatisch im `displaystyle` sein.

Math advanced

Mehr Mathe mit mathtools

L^AT_EX-Code:

```
\mathsf{sinc}(x) \coloneqq  
\begin{dcases}  
1 & x = 0 \\  
\frac{\sin(x)}{x} & \text{sonst}  
\end{dcases}
```

Ergebnis:

$$\text{sinc}(x) := \begin{cases} 1 & x = 0 \\ \frac{\sin(x)}{x} & \text{sonst} \end{cases}$$

Math advanced

Mehr Mathe mit mathtools

L^AT_EX-Code:

```
\underbrace{\exp(i x)}_{\text{Hier wird es hässlich}}
```

=

```
\cos(x) + \underbrace{i \sin(x)}_{\mathclap{\text{Hier wird es hübsch}}}
```

Ergebnis:

$$\underbrace{\exp(ix)}_{\text{Hier wird es hässlich}} = \cos(x) + \underbrace{i \sin(x)}_{\text{Hier wird es hübsch}}$$

Math advanced

SI-Einheiten mit siunitx

L^AT_EX-Code:

```
\sisetup{...} % Einstellungen für siunitx
```

```
Wissenschaftliche Notation: \(\text{n} = \text{num}\{1.1\text{e}3\}\) \\\
```

```
Einheiten: \(\text{J} = \text{kg}\,\text{m}^2\,\text{per}\,\text{s}^2\) \\\
```

```
Kombiniert: \(\text{SI}\{50\}\,\text{percent}\, c = \text{SI}\{1.5\text{e}8\}\,\text{m}\,\text{per}\,\text{s}\) \\\
```

```
Intervalle: \(\text{SIrange}\{1.3\text{e}5\}\text{--}\{3.6\text{e}6\}\,\text{kg}\,\text{m}\,\text{per}\,\text{s}\) \\\
```

Ergebnis:

Wissenschaftliche Notation: $n = 1,1 \cdot 10^3$

Einheiten: $J = \frac{\text{kg m}^2}{\text{s}^2}$

Kombiniert: $50\,\% c = 1,5 \cdot 10^8 \frac{\text{m}}{\text{s}}$

Intervalle: $1,3 \cdot 10^5 \text{--} 3,6 \cdot 10^6 \frac{\text{kg m}}{\text{s}}$

Math advanced

Noch mehr Mathe mit physics

L^AT_EX-Code:

Beträge und Normen:

`\(\norm{\vec{x}} = \sum_{k=1}^n \abs{x_k}^2 \) \\\`Landau-Notation: `\(\order{n \cdot \log(n)} \) \\\`Differentiale: `\(\int f(x) \dd{x};``\quad \dv{f}{x}; \quad \quad \pdv[2]{f}{x}\)`**Ergebnis:**Beträge und Normen: $\|\vec{x}\| = \sum_{k=1}^n |x_k|^2$ Landau-Notation: $\mathcal{O}(n \cdot \log(n))$ Differentiale: $\int f(x) dx; \quad \frac{df}{dx}; \quad \frac{\partial^2 f}{\partial x^2}$

TikZ

Grundlagen

TikZ

L^AT_EX-Code:

```
\begin{tikzpicture}
  \node (v) at (3,1) {$v$};

  \draw (0,0) -- (2,0) --
        (2,1) -- (v);
\end{tikzpicture}
```

Ergebnis:



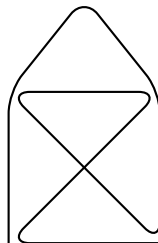
Grundlagen

TikZ

L^AT_EX-Code:

```
\begin{tikzpicture}
  \draw[thick,rounded corners=8pt]
    (0,0) -- (0,2) -- (1,3.25) --
    (2,2) -- (2,0) -- (0,2) --
    (2,2) -- (0,0) -- (2,0);
\end{tikzpicture}
```

Ergebnis:



Grundlagen

Nodes und Lines

TikZ-Code:

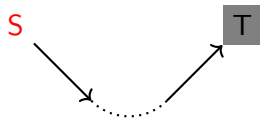
```
\begin{tikzpicture}
  \node[red]          (s) at (0, 0) {S};
  \node[fill=gray] (t) at (3, 0) {T};

  \draw[thick, ->]    (s)      -- (1, -1);
  \draw[thick, dotted]
    (1, -1) to [bend right = 45] (2, -1);
  \draw[thick,->]    (2, -1) -- (t);
\end{tikzpicture}
```


Grundlagen

Nodes und Lines

Ergebnis:



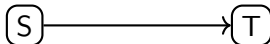
Grundlagen

Styles für gesamtes TikZpicture

TikZ-Code:

```
\begin{tikzpicture}
[
  ->, thick,
  knoten/.style={shape=rectangle,draw=black,rounded corners}
]
\node[knoten] (s) at (0, 0) {S};
\node[knoten] (t) at (3, 0) {T};
\draw[->] (s) -- (t);
\end{tikzpicture}
```

Ergebnis:



Automaten

Einführung

- Automaten (state machines) per automata-Paket
- Für Positionierung positioning-Paket
- Und für Pfeile arrows-Paket

Mehr Informationen über Automaten, Pfeile, Positionierung, Optionen, etc. gibt es unter
hauke-stieler.de/public/tikz-for-state-machines.pdf

(im selben Ordner ist auch die entsprechende *.tex Datei)

Automaten

Zustände

TikZ-Code:

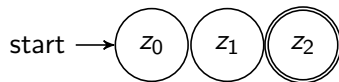
```
\usetikzlibrary{
    automata,
    arrows}
% ...
\begin{tikzpicture}[->,
    >=stealth',
    semithick]

    \node[state,initial]    (0)                {$z_0$};
    \node[state]            (1) [right of=0]    {$z_1$};
    \node[state,accepting] (2) [right of=1]    {$z_2$};
\end{tikzpicture}
```

Automaten

Zustände

Ergebnis:



Automaten

Positionierung

TikZ-Code:

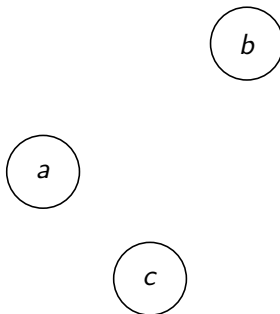
```
\usetikzlibrary{
    automata,
    arrows,
    positioning}
% ...
\begin{tikzpicture}[->,
    >=stealth',
    semithick,
    node distance=2cm]

    \node [state] (a)                                {$a$};
    \node [state] (b) [above right=1cm and 2cm of a] {$b$};
    \node [state] (c) [below right of = a]            {$c$};
\end{tikzpicture}
```

Automaten

Positionierung

Ergebnis:



Automaten

Pfeile

TikZ-Code:

```
\begin{tikzpicture}[->,
  >=stealth',
  semithick,
  node distance=2cm]

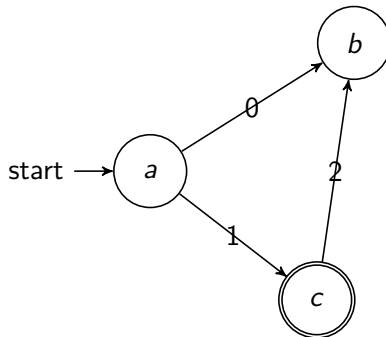
\node [state,initial] (a)           {$a$};
\node [state] (b)
  [above right=1cm and 2cm of a]     {$b$};
\node [state,accepting] (c)
  [below right = 1cm and 1.5cm of a] {$c$};

\path (a) edge node {0} (b)
      (a) edge node {1} (c)
      (c) edge node {2} (b);
\end{tikzpicture}
```

Automaten

Pfeile

Ergebnis:



Automaten

Pfeile

TikZ-Code:

```
\begin{tikzpicture}[->,
  >=stealth',
  semithick,
  node distance=2cm]

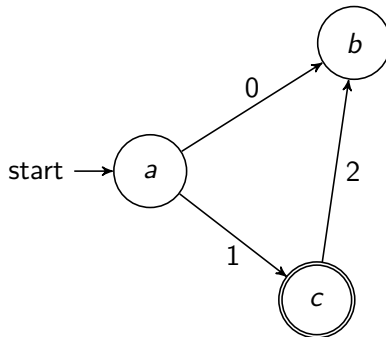
\node [state,initial]    (a)           {$a$};
\node [state]            (b)
  [above right=1cm and 2cm of a]      {$b$};
\node [state,accepting] (c)
  [below right = 1cm and 1.5cm of a] {$c$};

\path (a) edge[above] node {0} (b)
        edge[below] node {1} (c)
        (c) edge[right] node {2} (b);
\end{tikzpicture}
```

Automaten

Pfeile

Ergebnis:



Automaten

Pfeile

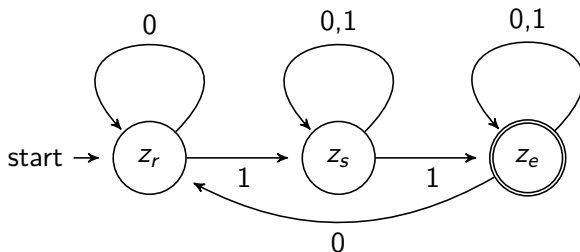
TikZ-Code:

```
\begin{tikzpicture}[->,>=stealth',  
  shorten >=5pt,  
  node distance=2.5cm,  
  semithick]  
  
  \node[initial,state]      (R)                {$z_r$};  
  \node[state]              (S) [right of=R]    {$z_s$};  
  \node[state,accepting]    (E) [right of=S]    {$z_e$};  
  
  \path (R) edge [loop,above] node {0} (R)  
         edge [below] node {1} (S)  
        (S) edge [loop,above] node {0,1} (S)  
         edge [below] node {1} (E)  
        (E) edge [bend left,below] node {0} (R)  
         edge [loop,above] node {0,1} (E);  
  
\end{tikzpicture}
```

Automaten

Pfeile

Ergebnis:



Funktionen Zeichnen

TikZ

```
\usepackage{pgf}
% ...
\begin{tikzpicture}[>=latex,semithick,font=\scriptsize,scale=0.75]
  \draw[very thin,color=lightgray] (-3.2,-1.2) grid (3.2,4.2);
  \draw[->] (-3.2,0) -- (3.4,0) node[right] {$x$};
  \draw[->] (0,-1.2) -- (0,4.4) node[above] {$y$};

  \foreach \x/\xtext in {-3/-3, -2/-2, -1/-1, 1/1, 2/2, 3/3}
  \draw[shift={(\x,0)}] (0pt,2pt) -- (0pt,-2pt) node[below] {$\xtext$};

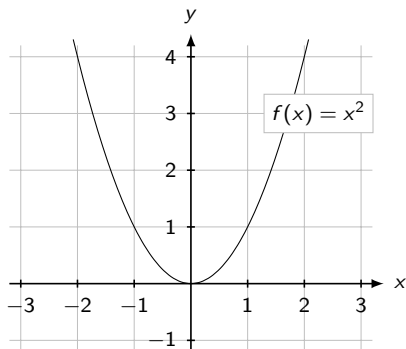
  \foreach \y/\ytext in {-1/-1, 1/1, 2/2, 3/3, 4/4}
  \draw[shift={(0,\y)}] (2pt,0pt) -- (-2pt,0pt) node[left] {$\ytext$};

  \draw[thin,domain=-2.075:2.075,smooth,variable=\x,black]
    plot ({\x},{\x*\x});
  \draw[thin] node[inner sep=1mm, fill=white, draw=lightgray]
    at (2.25,3) {$f(x)=x^2$};
\end{tikzpicture}
```

Funktionen Zeichnen

TikZ

Ergebnis:



Alternative: Gnuplot lässt sich vielfältig mit L^AT_EX kombinieren

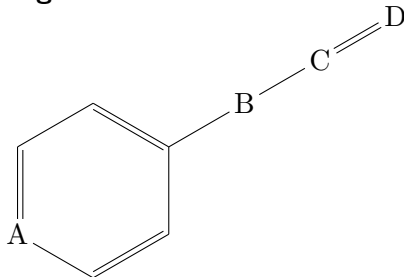
Was sonst so geht

Chemische Bilder mit chemfig

L^AT_EX-Code:

```
\chemfig{A*6(---(-B-C=D)---)}
```

Ergebnis:



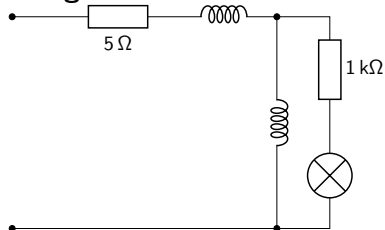
Was sonst so geht

Schaltskizzen mit CircuiTikZ

L^AT_EX-Code:

```
\begin{circuitikz}[european,cute inductors]
\draw (0,0) to [short, *-] (6,0)
      to [lamp] (6,2)
      to [R, l_= $1\text{k}\Omega$ ] (6,4)
      to [short] (5,4)
      (0,4) to [short, *-] (1,4)
      to [R, l_= $5\Omega$ ] (3,4)
      to [L] (5,4)
      to [L, *-] (5,0);
\end{circuitikz}
```

Ergebnis:



Was sonst so geht

Petrinetze

L^AT_EX-Code:

```

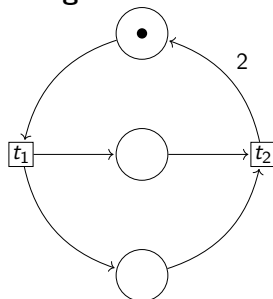
\usetikzlibrary{petri}
\begin{tikzpicture}[node distance=1.5cm,auto]
  \node [place,tokens=1] (p1) {};
  \node [place] (p2) [below of=p1] {};
  \node [place] (p3) [below of=p2] {};

  \node [transition] (t1) [left of=p2] {$t_1$}
    edge [pre,bend left] (p1)
    edge [post,bend right] (p3)
    edge [post] (p2);

  \node [transition] (t2) [right of=p2] {$t_2$}
    edge [pre] (p2)
    edge [pre,bend left] (p3)
    edge [post,bend right] node[swap] {2} (p1);
\end{tikzpicture}

```

Ergebnis:



Versionsverwaltung

Version control in a nutshell

Versionsverwaltungs-Software (VCS):

- Ermöglicht es, an vielen Rechnern Dateien kollaborativ zu bearbeiten
- Über Branches könnt ihr parallel an den gleichen Dateien arbeiten
- Branches können zusammengeführt werden (merge)
- Vllt. seht ihr noch ein KBS dazu ;)

Git:

- Das populärste VCS von Linus Torvalds
- Unter Linux meist vorinstalliert, unter Windows z.B. via git-scm
- Dezentral, vielseitig, einfach, mächtig

Versionsverwaltung

.gitignore

- Die .gitignore verwaltet automatisch ignorierte Dateien im Repository
- L^AT_EX erzeugt viele temporäre Dateien
- The lazy way: <https://github.com/github/gitignore>

<https://raw.githubusercontent.com/github/gitignore/master/TeX.gitignore>

Versionsverwaltung

.gitignore

Beispiel:

```
## Core latex/pdflatex auxiliary files:
```

```
*.aux
```

```
*.log
```

```
[...]
```

```
## Intermediate documents:
```

```
*.dvi
```

```
*-converted-to.*
```

```
# *.ps
```

```
# *.pdf
```

```
[...]
```

Versionsverwaltung

make & latexmk

make:

- make automatisiert das Bauen von Projekten
- Macht Dokumentation meistens überflüssig
- Automatischer Abgleich von Änderungsdatum von Quelle und Ziel
- Festlegen des Build-Prozesses durch Regeln in der Makefile
- Regelfindung per Pattern-Matching

latexmk:

- Kann L^AT_EX-Dokumente vollautomatisch kompilieren
- Alternative: `latexrun`

Versionsverwaltung

make & latexmk

Beispiel für Makefile:

```
.PHONY: default all clean
```

```
default: all
```

```
all: Abgabe.pdf
```

```
%.pdf: %.tex
```

```
latexmk -pdf $<
```

```
clean:
```

```
latexmk -C
```