

# Das Textsatzsystem $\text{\LaTeX}$

Carsten Geckeler

Linux User Group Tübingen

6. Juni 2006

# Übersicht

- 1 Einführung in T<sub>E</sub>X/L<sub>A</sub>T<sub>E</sub>X
  - Einleitung
  - Erste Schritte in L<sub>A</sub>T<sub>E</sub>X
  - Weitere L<sub>A</sub>T<sub>E</sub>X-Elemente
  
- 2 Präsentationen mit BEAMER
  - Einleitung
  - Themes
  - Dynamische Effekte

# Geschichte von T<sub>E</sub>X und L<sup>A</sup>T<sub>E</sub>X

- 1968 Donald E. Knuth beginnt mit seinem mehrbändig geplanten Buch *The Art of Computer Programming*
- 1977 Knuth beginnt mit der Arbeit an T<sub>E</sub>X wegen seiner Unzufriedenheit mit der Satzqualität seines Buches
- 1982 T<sub>E</sub>X 82 wird veröffentlicht
- 1985 das Makropaket L<sup>A</sup>T<sub>E</sub>X von Leslie Lamport erscheint in Version 2.09
- 1989 T<sub>E</sub>X wird eingefroren
- 1994 L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> erscheint als Nachfolger von L<sup>A</sup>T<sub>E</sub>X 2.09

## Aktuelle Versionen:

- T<sub>E</sub>X 3.14159
- METAFONT 2.7182
- L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> Dec. 2005

# Vor- und Nachteile von $\text{\LaTeX}$

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

# Vor- und Nachteile von $\text{\LaTeX}$

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten

# Logisches Markup

In der Quelldatei beschreibt der Autor die logische Struktur des Dokuments.

# Logisches Markup

In der Quelldatei beschreibt der Autor die logische Struktur des Dokuments.

Das heißt, man teilt dem Programm mit, ...

- was eine Überschrift ist.
- was hervorgehoben werden soll.
- wer der Autor ist.
- wie der Dokumententitel lautet.
- welche Abbildung mit welcher Grafik eingefügt werden soll.
- ...

# Logisches Markup

In der Quelldatei beschreibt der Autor die logische Struktur des Dokuments.

Das heißt, man teilt dem Programm mit, ...

- was eine Überschrift ist.
- was hervorgehoben werden soll.
- wer der Autor ist.
- wie der Dokumententitel lautet.
- welche Abbildung mit welcher Grafik eingefügt werden soll.
- ...

$\text{\LaTeX}$ , die Dokumentklasse und die Pakete entscheiden, *wie* der Text gesetzt wird.



# Unser erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

Ein sehr einfaches Dokument:

`bsp1.tex`

```
\documentclass{scrartcl}  
% ein Kommentar  
\begin{document}  
Hello \LaTeX\ World!  
\end{document}
```

*Die grobe Struktur:*

# Unser erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

Ein sehr einfaches Dokument:

bsp1.tex

```
\documentclass{scrartcl}
% ein Kommentar
\begin{document}
Hello \LaTeX\ World!
\end{document}
```

*Die grobe Struktur:*

- die Dokumentenklasse

# Unser erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

Ein sehr einfaches Dokument:

bsp1.tex

```
\documentclass{scrartcl}  
% ein Kommentar  
\begin{document}  
Hello \LaTeX\ World!  
\end{document}
```

*Die grobe Struktur:*

- die Dokumentenklasse
- die Präambel

# Unser erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

Ein sehr einfaches Dokument:

```
bsp1.tex
```

```
\documentclass{scrartcl}  
% ein Kommentar  
\begin{document}  
Hello \LaTeX\ World!  
\end{document}
```

*Die grobe Struktur:*

- die Dokumentenklasse
- die Präambel
- **der Text**

# Unser erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

Ein sehr einfaches Dokument:

```
bsp1.tex
```

```
\documentclass{scrartcl}
% ein Kommentar
\begin{document}
Hello \LaTeX\ World!
\end{document}
```

*Die grobe Struktur:*

- die Dokumentenklasse
- die Präambel
- der Text

*Befehle:*

- **Makros**

# Unser erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

Ein sehr einfaches Dokument:

```
bsp1.tex
```

```
\documentclass{scrartcl}  
% ein Kommentar  
\begin{document}  
Hello \LaTeX\ World!  
\end{document}
```

*Die grobe Struktur:*

- die Dokumentenklasse
- die Präambel
- der Text

*Befehle:*

- Makros
- **Argumente**

# Unser erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

Ein sehr einfaches Dokument:

```
bsp1.tex
```

```
\documentclass{scrartcl}  
% ein Kommentar  
\begin{document}  
Hello \LaTeX\ World!  
\end{document}
```

*Die grobe Struktur:*

- die Dokumentenklasse
- die Präambel
- der Text

*Befehle:*

- Makros
- Argumente
- **Umgebungen**

# Unser erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

Ein sehr einfaches Dokument:

```
bsp1.tex
```

```
\documentclass{scrartcl}  
% ein Kommentar  
\begin{document}  
Hello \LaTeX\ World!  
\end{document}
```

*Die grobe Struktur:*

- die Dokumentenklasse
- die Präambel
- der Text

*Befehle:*

- Makros
- Argumente
- Umgebungen
- **Kommentare**



# Vom Quelltext zur Ausgabe

`latex bsp1` generiert aus der Quelldatei `bsp1.tex` eine  
Ausgabedatei `bsp1.dvi` (**d**evice **i**ndependent)

# Vom Quelltext zur Ausgabe

`latex bsp1` generiert aus der Quelldatei `bsp1.tex` eine  
Ausgabedatei `bsp1.dvi` (**d**evice **i**ndependent)

Die DVI-Datei lässt sich mit vielen Treiber auf das gewünschte  
Ausgabegerät ausgeben.

`xdvi bsp1` zeigt das  $\text{\LaTeX}$ -Dokument auf dem Bildschirm an

`dvips bsp1` erzeugt eine PostScript-Datei `bsp1.ps`

# Vom Quelltext zur Ausgabe

`latex bsp1` generiert aus der Quelldatei `bsp1.tex` eine  
Ausgabedatei `bsp1.dvi` (**d**evice **i**ndependent)

Die DVI-Datei lässt sich mit vielen Treiber auf das gewünschte  
Ausgabegerät ausgeben.

`xdvi bsp1` zeigt das  $\text{\LaTeX}$ -Dokument auf dem Bildschirm an

`dvips bsp1` erzeugt eine PostScript-Datei `bsp1.ps`

Inzwischen gibt es auch `pdflatex`:

`pdflatex bsp1` generiert aus der Quelldatei `bsp1.tex` direkt eine  
PDF-Datei `bsp1.pdf`

Hello L<sup>A</sup>T<sub>E</sub>X World!

# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,
  12pt,halparskip*]{scrartcl}
\usepackage{ngerman}
\usepackage[latin1]{inputenc}
\setlength{\columnsep}{2em}
\author{Carsten Geckeler}
\title{Ein kleiner Beispieltext}
\begin{document}
\maketitle
\tableofcontents
\section{Erster Abschnitt} ...
\subsection{Ein Unterabschnitt} ...
\subsection{Noch ein Unterabschnitt} ...
\section{Der zweite Abschnitt} ...
\end{document}
```

*Neue Elemente:*

# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,  
12pt,halparskip*]{scrartcl}  
\usepackage{ngerman}  
\usepackage[latin1]{inputenc}  
\setlength{\columnsep}{2em}  
\author{Carsten Geckeler}  
\title{Ein kleiner Beispieltext}  
\begin{document}  
\maketitle  
\tableofcontents  
\section{Erster Abschnitt} ...  
\subsection{Ein Unterabschnitt} ...  
\subsection{Noch ein Unterabschnitt} ...  
\section{Der zweite Abschnitt} ...  
\end{document}
```

*Neue Elemente:*

- Dokumentenoptionen

# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,
  12pt,halparskip*]{scrartcl}
\usepackage{ngerman}
\usepackage[latin1]{inputenc}
\setlength{\columnsep}{2em}
\author{Carsten Geckeler}
\title{Ein kleiner Beispieltext}
\begin{document}
\maketitle
\tableofcontents
\section{Erster Abschnitt} ...
\subsection{Ein Unterabschnitt} ...
\subsection{Noch ein Unterabschnitt} ...
\section{Der zweite Abschnitt} ...
\end{document}
```

## Neue Elemente:

- Dokumentenoptionen
- **Laden von Paketen**

# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,
  12pt,halparskip*]{scrartcl}
\usepackage{ngerman}
\usepackage[latin1]{inputenc}
\setlength{\columnsep}{2em}
\author{Carsten Geckeler}
\title{Ein kleiner Beispieltext}
\begin{document}
\maketitle
\tableofcontents
\section{Erster Abschnitt} ...
\subsection{Ein Unterabschnitt} ...
\subsection{Noch ein Unterabschnitt} ...
\section{Der zweite Abschnitt} ...
\end{document}
```

## Neue Elemente:

- Dokumentenoptionen
- Laden von Paketen
- **optionale Argumente**



# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,
  12pt,halparskip*]{scrartcl}
\usepackage{ngerman}
\usepackage[latin1]{inputenc}
\setlength{\columnsep}{2em}
\author{Carsten Geckeler}
\title{Ein kleiner Beispieltext}
\begin{document}
\maketitle
\tableofcontents
\section{Erster Abschnitt} ...
\subsection{Ein Unterabschnitt} ...
\subsection{Noch ein Unterabschnitt} ...
\section{Der zweite Abschnitt} ...
\end{document}
```

## Neue Elemente:

- Dokumentenoptionen
- Laden von Paketen
- optionale Argumente
- **Strukturbefehle**

# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,
  12pt,halparskip*]{scrartcl}
\usepackage{ngerman}
\usepackage[latin1]{inputenc}
\setlength{\columnsep}{2em}
\author{Carsten Geckeler}
\title{Ein kleiner Beispieltext}
\begin{document}
\maketitle
\tableofcontents
\section{Erster Abschnitt} ...
\subsection{Ein Unterabschnitt} ...
\subsection{Noch ein Unterabschnitt} ...
\section{Der zweite Abschnitt} ...
\end{document}
```

## Neue Elemente:

- Dokumentenoptionen
- Laden von Paketen
- optionale Argumente
- Strukturbefehle
- **Autor und Titel**

# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,
  12pt,halparskip*]{scrartcl}
\usepackage{ngerman}
\usepackage[latin1]{inputenc}
\setlength{\columnsep}{2em}
\author{Carsten Geckeler}
\title{Ein kleiner Beispieltext}
\begin{document}
\maketitle
\tableofcontents
\section{Erster Abschnitt} ...
\subsection{Ein Unterabschnitt} ...
\subsection{Noch ein Unterabschnitt} ...
\section{Der zweite Abschnitt} ...
\end{document}
```

## Neue Elemente:

- Dokumentenoptionen
- Laden von Paketen
- optionale Argumente
- Strukturbefehle
- Autor und Titel
- **Ausgabe des Titels**

# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,
  12pt,halparskip*]{scrartcl}
\usepackage{ngerman}
\usepackage[latin1]{inputenc}
\setlength{\columnsep}{2em}
\author{Carsten Geckeler}
\title{Ein kleiner Beispieltext}
\begin{document}
\maketitle
\tableofcontents
\section{Erster Abschnitt} ...
\subsection{Ein Unterabschnitt} ...
\subsection{Noch ein Unterabschnitt} ...
\section{Der zweite Abschnitt} ...
\end{document}
```

## Neue Elemente:

- Dokumentenoptionen
- Laden von Paketen
- optionale Argumente
- Strukturbefehle
- Autor und Titel
- Ausgabe des Titels
- **Inhaltsverzeichnis**

# Das L<sup>A</sup>T<sub>E</sub>X-Dokument wächst

## bsp2.tex

```
\documentclass[a4paper,twocolumn,
  12pt,halparskip*]{scrartcl}
\usepackage{ngerman}
\usepackage[latin1]{inputenc}
\setlength{\columnsep}{2em}
\author{Carsten Geckeler}
\title{Ein kleiner Beispieltext}
\begin{document}
\maketitle
\tableofcontents
\section{Erster Abschnitt} ...
\subsection{Ein Unterabschnitt} ...
\subsection{Noch ein Unterabschnitt} ...
\section{Der zweite Abschnitt} ...
\end{document}
```

## Neue Elemente:

- Dokumentenoptionen
- Laden von Paketen
- optionale Argumente
- Strukturbefehle
- Autor und Titel
- Ausgabe des Titels
- Inhaltsverzeichnis
- größerer  
Spaltenabstand

# Ein kleiner Beispieltext

Carsten Geckeler

8. Juni 2006

## Inhaltsverzeichnis

<b>1 Erster Abschnitt</b>	<b>1</b>
1.1 Ein Unterabschnitt . . . .	1
1.2 Noch ein Unterabschnitt .	1
<b>2 Der zweite Abschnitt</b>	<b>1</b>

## 1 Erster Abschnitt

Weit hinten, hinter den Wortbergen, fern der Länder Vokalien und Konsonantien leben die Blindtexte. Abgeschieden wohnen sie in Buchstabhausen an der Küste des Semantik, eines großen Sprachozeans.

### 1.1 Ein Unterabschnitt

Ein kleines Bächlein namens Duden fließt durch ihren Ort und versorgt sie mit den nötigen Regelialien. Es ist ein paradiesmatisches Land, in dem einem gebratene Satzteile in den Mund fliegen.

### 1.2 Noch ein Unterabschnitt

Nicht einmal von der allmächtigen Interpunktion werden die Blindtexte beherrscht – ein geradezu unorthographisches Leben. Eines Tages aber beschloß eine kleine Zeile Blindtext, ihr Name war

Lorem Ipsum, hinaus zu gehen in die weite Grammatik.

## 2 Der zweite Abschnitt

Der große Oxmox riet ihr davon ab, da es dort wimmele von bösen Kommata, wilden Fragezeichen und hinterhältigen Semikola, doch das Blindtextchen ließ sich nicht beirren. Es packte seine sieben Versalien, schob sich sein Initial in den Gürtel und machte sich auf den Weg. Als es die ersten Hügel des Kursivgebirges erklimmen hatte, warf es einen letzten Blick zurück auf die Skyline seiner Heimatstadt Buchstabhausen, die Headline von Alphabetdorf und die Subline seiner eigenen Straße, der Zeilengasse.

# Dokumentenklassen und Strukturbefehle

weitere Dokumentenklassen

`scrbook`, `scrartcl`, `sclttr2`, `beamer`, `dinbrief`, ...

# Dokumentenklassen und Strukturbefehle

## weitere Dokumentenklassen

`scrbook`, `scrartcl`, `scrlettr2`, `beamer`, `dinbrief`, ...

Die KOMA-Skriptklassen (`scr...`) sollten statt den früheren Standardklassen (`article`, ...) benutzt werden. Vorteile:

- bessere Dokumentation
- einfacher zu Konfigurieren



# Dokumentenklassen und Strukturbefehle

## weitere Dokumentenklassen

`scrbook`, `scrartcl`, `scrlettr2`, `beamer`, `dinbrief`, ...

Die KOMA-Skriptklassen (`scr...`) sollten statt den früheren Standardklassen (`article`, ...) benutzt werden. Vorteile:

- bessere Dokumentation
- einfacher zu Konfigurieren

## Strukturbefehle für Kapitel, Abschnitte

```
\part{...}, \chapter{...}, \section{...},  
\subsection{...}, \subsubsection{...},  
\paragraph{...}, \subparagraph{...}
```

Nicht alle Strukturbefehle sind bei allen Klassen verfügbar (z. B. `part` und `chapter` nicht bei `scrartcl`).

# Verzeichnisse und Schriften

## Verzeichnisse

```
\tableofcontents, \listoffigures, \listoftables
```

Zum Erstellen eines Inhalts-, Abbildungs- und Tabellenverzeichnisses.

# Verzeichnisse und Schriften

## Verzeichnisse

```
\tableofcontents, \listoffigures, \listoftables
```

Zum Erstellen eines Inhalts-, Abbildungs- und Tabellenverzeichnisses.

## Änderungen der Schrift

```
\textbf{...}, \textit{...}, \textsl{...}  
\tiny, \small, \large, \LARGE, \Huge
```

**fette**, *kursive* und *geneigte* Schrift.

winzige, kleine, große, **größere** und **riesige** Schrift.

# Umgebungen

Umgebungen mit `\begin{...}` und `\end{...}`

`figure`, `table`, `tabular`, `itemize`, `emumerate`, `quote`,  
`center`, `verbatim`, `frame`, `bibliography`, ...

# Umgebungen

## Umgebungen mit `\begin{...}` und `\end{...}`

`figure`, `table`, `tabular`, `itemize`, `enumerate`, `quote`,  
`center`, `verbatim`, `frame`, `bibliography`, ...

Als Beispiel eine `enumerate`-Umgebung:

```
\begin{enumerate}
\item erster Punkt
\item zweiter Punkt
\end{enumerate}
```

# Umgebungen

## Umgebungen mit `\begin{...}` und `\end{...}`

`figure`, `table`, `tabular`, `itemize`, `enumerate`, `quote`,  
`center`, `verbatim`, `frame`, `bibliography`, ...

Als Beispiel eine `enumerate`-Umgebung:

```
\begin{enumerate}
\item erster Punkt
\item zweiter Punkt
\end{enumerate}
```

ergibt:

- 1 erster Punkt
- 2 zweiter Punkt

# Anforderungen an eine Präsentation

## *Anforderungen an eine Präsentation:*

- Ausgabe in PDF  
→ Acrobat Reader auf vielen Plattformen erhältlich
- ansprechendes Design
- einfache Konfiguration (z. B. Themes)
- sinnvolle Navigationselemente
- dynamische Effekte  
→ schrittweises Aufdecken bestimmter Textteile
- mögliche Ausgabe von Handouts

# Anforderungen an eine Präsentation

## *Anforderungen an eine Präsentation:*

- Ausgabe in PDF  
→ Acrobat Reader auf vielen Plattformen erhältlich
- ansprechendes Design
- einfache Konfiguration (z. B. Themes)
- sinnvolle Navigationselemente
- dynamische Effekte  
→ schrittweises Aufdecken bestimmter Textteile
- mögliche Ausgabe von Handouts

Es gibt einige gute Klassen für Präsentationen (z. B. `slides`, `FoilTeX`, `powerdot`), hier soll aber nur die BEAMER-Klasse behandelt werden.



# Grundgerüst einer Beamer-Präsentation

```
\documentclass{beamer}
\usepackage[latin1]{inputenc}
\usetheme{Copenhagen}
\begin{document}

\author{Carsten Geckeler} \title{Das Textsatzsystem \LaTeX}
\institute{Linux User Group Tübingen} \date{6.~Juni 2006}

\frame{\titlepage}

\begin{frame}
  \frametitle{Übersicht}
  \tableofcontents
\end{frame}

...
\end{document}
```

# Themes

BEAMER bietet eine einfache Möglichkeit, das Design der Präsentation via Themes zu ändern:

## Ändern des Presentation Theme

```
\usetheme{theme}
```

# Themes

BEAMER bietet eine einfache Möglichkeit, das Design der Präsentation via Themes zu ändern:

## Ändern des Presentation Theme

```
\usetheme{theme}
```

Man kann aber auch nur Teilbereiche via Themes beeinflussen:

## Themes für Struktur, Aussehen, Farben, Schriften

```
\useoutertheme{theme}
```

```
\useinnertheme{theme}
```

```
\usecolortheme{theme}
```

```
\usefonttheme{theme}
```

# Themes

BEAMER bietet eine einfache Möglichkeit, das Design der Präsentation via Themes zu ändern:

## Ändern des Presentation Theme

```
\usetheme{theme}
```

Man kann aber auch nur Teilbereiche via Themes beeinflussen:

## Themes für Struktur, Aussehen, Farben, Schriften

```
\useoutertheme{theme}
```

```
\useinnertheme{theme}
```

```
\usecolortheme{theme}
```

```
\usefonttheme{theme}
```

Im Folgenden kommen Beispiele für Themes ...

## *Vorteile:*

- ▶ L<sup>A</sup>T<sub>E</sub>X liefert einen professionellen Textsatz
- ▶ Trennung von generischem/logischem und visuellem Markup
- ▶ umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- ▶ auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- ▶ kostenlos

## *Nachteile:*

- ▶ für Einsteiger sehr ungewohnte Eingabe
- ▶ Änderungen des Layouts teilweise recht kompliziert
- ▶ schwierige Installation neuer Schriftarten

# Vor- und Nachteile von $\text{\LaTeX}$

Warsaw

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten

# Vor- und Nachteile von L<sup>A</sup>T<sub>E</sub>X

## Montpellier

### *Vorteile:*

- ▶ L<sup>A</sup>T<sub>E</sub>X liefert einen professionellen Textsatz
- ▶ Trennung von generischem/logischem und visuellem Markup
- ▶ umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- ▶ auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- ▶ kostenlos

### *Nachteile:*

- ▶ für Einsteiger sehr ungewohnte Eingabe
- ▶ Änderungen des Layouts teilweise recht kompliziert
- ▶ schwierige Installation neuer Schriftarten

## Vorteile

- *L<sup>A</sup>T<sub>E</sub>X liefert einen professionellen Textsatz*
- *Trennung von generischem/logischem und visuellem Markup*
- *umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen*
- *auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis*
- *kostenlos*

## Nachteile

- *für Einsteiger sehr ungewohnte Eingabe*
- *Änderungen des Layouts teilweise recht kompliziert*
- *schwierige Installation neuer Schriftarten*



# Overlays mit `\pause`

Mit dem Befehl `\pause` wird eine Unterbrechung eingefügt. Dieser Effekt wird durch mehrere Seiten im PDF erzeugt, wobei die ersten nur einen Teil des Textes beinhalten.

```
\pause
```

# Overlays mit `\pause`

Mit dem Befehl `\pause` wird eine Unterbrechung eingefügt. Dieser Effekt wird durch mehrere Seiten im PDF erzeugt, wobei die ersten nur einen Teil des Textes beinhalten.

```
\pause
```

Die nächsten Seiten zeigen dann den weiteren Text.

```
\pause
```

# Overlays mit `\pause`

Mit dem Befehl `\pause` wird eine Unterbrechung eingefügt. Dieser Effekt wird durch mehrere Seiten im PDF erzeugt, wobei die ersten nur einen Teil des Textes beinhalten.

```
\pause
```

Die nächsten Seiten zeigen dann den weiteren Text.

```
\pause
```

Man kann beliebig viele `\pause`-Befehle einfügen.

# Einfache dynamische Effekte

Teile des Textes können auf bestimmten Overlay sichtbar gemacht werden und in ihrem Erscheinen geändert werden.

```
\alert<1>{Dieser Text wird hervorgehoben.}  
\uncover<2->{Dieser Text erscheint später.}  
{\color<3>[rgb]{0,0,1}Dieser Text ist mal blau.}  
\only<1>{Dieser Text verschwindet komplett.}  
\textbf<1,3>{Dieser Text ist manchmal fett.}
```

Dieser Text wird hervorgehoben.

Dieser Text ist mal blau.

Dieser Text verschwindet komplett.

**Dieser Text ist manchmal fett.**

# Einfache dynamische Effekte

Teile des Textes können auf bestimmten Overlay sichtbar gemacht werden und in ihrem Erscheinen geändert werden.

```
\alert<1>{Dieser Text wird hervorgehoben.}  
\uncover<2->{Dieser Text erscheint später.}  
{\color<3>[rgb]{0,0,1}Dieser Text ist mal blau.}  
\only<1>{Dieser Text verschwindet komplett.}  
\textbf<1,3>{Dieser Text ist manchmal fett.}
```

Dieser Text wird hervorgehoben.

Dieser Text erscheint später.

Dieser Text ist mal blau.

Dieser Text ist manchmal fett.

# Einfache dynamische Effekte

Teile des Textes können auf bestimmten Overlay sichtbar gemacht werden und in ihrem Erscheinen geändert werden.

```
\alert<1>{Dieser Text wird hervorgehoben.}  
\uncover<2->{Dieser Text erscheint später.}  
{\color<3>[rgb]{0,0,1}Dieser Text ist mal blau.}  
\only<1>{Dieser Text verschwindet komplett.}  
\textbf<1,3>{Dieser Text ist manchmal fett.}
```

Dieser Text wird hervorgehoben.

Dieser Text erscheint später.

Dieser Text ist mal blau.

**Dieser Text ist manchmal fett.**

# Kompliziertere dynamische Effekte

BEAMER erlaubt sehr flexible Definitionen:

```
\begin{itemize}[<+- | alert@+>]
\item ...
...
\end{itemize}
```

# Kompliziertere dynamische Effekte

BEAMER erlaubt sehr flexible Definitionen:

```
\begin{itemize}[<+- | alert@+>]
\item ...
...
\end{itemize}
```

Außerdem kann man den verdeckten Text transparent durchscheinen lassen:

```
\setbeamercovered{highly dynamic}
```



# Kompliziertere dynamische Effekte

BEAMER erlaubt sehr flexible Definitionen:

```
\begin{itemize}[<+- | alert@+>]
\item ...
...
\end{itemize}
```

Außerdem kann man den verdeckten Text transparent durchscheinen lassen:

```
\setbeamercovered{highly dynamic}
```

Das folgende Beispiel verdeutlicht beide Methoden.

# Vor- und Nachteile von $\LaTeX$ (nochmal)

## *Vorteile:*

- $\LaTeX$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten

# Vor- und Nachteile von $\text{\LaTeX}$ (nochmal)

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- **Trennung von generischem/logischem und visuellem Markup**
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten

# Vor- und Nachteile von $\text{\LaTeX}$ (nochmal)

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten

# Vor- und Nachteile von $\text{\LaTeX}$ (nochmal)

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten

# Vor- und Nachteile von $\text{\LaTeX}$ (nochmal)

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- **kostenlos**

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten

# Vor- und Nachteile von $\text{\LaTeX}$ (nochmal)

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten

# Vor- und Nachteile von $\text{\LaTeX}$ (nochmal)

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- schwierige Installation neuer Schriftarten



# Vor- und Nachteile von $\text{\LaTeX}$ (nochmal)

## *Vorteile:*

- $\text{\LaTeX}$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- **Änderungen des Layouts teilweise recht kompliziert**
- schwierige Installation neuer Schriftarten

# Vor- und Nachteile von $\LaTeX$ (nochmal)

## *Vorteile:*

- $\LaTeX$  liefert einen professionellen Textsatz
- Trennung von generischem/logischem und visuellem Markup
- umfangreiche Sammlung von Klassen und Paketen für unterschiedlichste Aufgaben und Anforderungen
- auf nahezu jeder Computerplattform erhältlich und liefert auch immer exakt das gleiche Ergebnis
- kostenlos

## *Nachteile:*

- für Einsteiger sehr ungewohnte Eingabe
- Änderungen des Layouts teilweise recht kompliziert
- **schwierige Installation neuer Schriftarten**

# Dokumentationen für $\text{\LaTeX}$ und Beamer

*Für  $\text{\LaTeX}$  allgemein:*

- Die  $\text{\LaTeX}$  2<sub>ε</sub>-Kurzbeschreibung  
<ftp://dante.ctan.org/tex-archive/info/lshort/german/l2kurz.pdf>
- KOMA-Skript-Dokumentation  
<ftp://dante.ctan.org/tex-archive/macros/latex/contrib/koma-script/scrguide.pdf>

*Für BEAMER:*

- BEAMER-Homepage bei Sourceforge  
<http://latex-beamer.sourceforge.net/>

Vielen Dank für Eure Aufmerksamkeit.