

Ausgewählte Kapitel der Systemsoftware (AKSS)

Arbeitstechnik

29. Oktober 2019

Timo Hönig, Laura Lawniczak, Tobias Langer

Lehrstuhl für Informatik 4
Friedrich-Alexander-Universität Erlangen-Nürnberg



Lehrstuhl für Verteilte Systeme
und Betriebssysteme



FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

Motivation

- Schreiben ist wichtiger Teil der wissenschaftlichen Arbeit
 - Präsentation der Erkenntnisse
 - ...nach etablierten Standards
- Herausforderungen beim Schreiben
 - Präsentation der Inhalte
 - Formatierung und Präsentation von Texten und Bildern
 - Verwaltung von Referenzen
 - Kollaboration mit KollegInnen

- Schreiben ist wichtiger Teil der wissenschaftlichen Arbeit
 - Präsentation der Erkenntnisse
 - ...nach etablierten Standards
- Herausforderungen beim Schreiben
 - Präsentation der Inhalte
 - **Formatierung und Präsentation von Texten und Bildern**
 - **Verwaltung von Referenzen**
 - **Kollaboration mit KollegInnen**

Thema heute

Werkzeuge für den wissenschaftlichen Schreibprozess

Agenda

LaTeX

LaTeX Beamer

Versionskontrollsysteme

Zusammenfassung

L^AT_EX

- L^AT_EX vereinfacht die Benutzung von T_EX
- TeX entwickelt von Donald E. Knuth (1970er Jahre)
- LaTeX entwickelt von Leslie Lamport (**Lamport TeX**, 1980er Jahre)
- Erweiterbar durch zahllose Pakete

■ Fettschrift

```
01 Hallo \textbf{Welt}.
```

→ Hallo **Welt**.

■ Kursivschrift

```
01 Hallo \textit{Welt}.
```

→ Hallo *Welt*.

■ Festbreitenschrift

```
01 Hallo \texttt{Welt}.
```

→ Hallo Welt.

■ Hervorgehobener Text

```
01 Hallo \emph{Welt}.
```

→ Hallo *Welt*.

- Neues Kapitel erstellen (u.U. optional)

```
01 \chapter{Kapitelüberschrift}
```

- Neue Unterkapitel erstellen

```
01 \section{Überschrift}
02 \subsection{Überschrift}
03 \subsubsection{Überschrift}
```

- Automatisch Inhaltsverzeichnis erstellen

```
01 \tableofcontents
```

- Paket: `graphicx`
- Einbinden von Bildern (z.B. Abbildung 1)

```
01 \begin{figure}  
02   \includegraphics[width=0.3\textwidth]{img/git-logo}  
03   \caption{Git Logo}  
04   \label{fig:git-logo}  
05 \end{figure}
```



Abbildung 1: Git Logo

■ ohne Nummerierung

```
01 \begin{itemize}
02   \item Erster Punkt
03   \item Zweiter Punkt
04 \end{itemize}
```

- Erster Punkt
- Zweiter Punkt

■ mit Nummerierung

```
01 \begin{enumerate}
02   \item Erster Punkt
03   \item Zweiter Punkt
04 \end{enumerate}
```

1. Erster Punkt
2. Zweiter Punkt

- Anlegen eines Referenzpunkts (z.B. eines Kapitels)

```
01 \section{Background}
02 \label{sec:background}
03 This section introduces important background knowledge.
04 [...]
```

- Spätere Referenzierung der Kapitelnummer oder Seite

```
01 As described in Section~\ref{sec:background} on
02 page~\pageref{sec:background}
03 [...]
```

```
01 As described in Section 2 on page 8
02 [...]
```

- Anlegen eines Referenzpunkts für ein Bild

```
01 \begin{figure}  
02     \includegraphics[width=0.3\textwidth]{img/git-logo}  
03     \caption{Git Logo}  
04     \label{fig:git-logo}  
05 \end{figure}
```

- Analog für
 - Tabellen
 - Gleichungen
 - ...

■ Verwaltung in einer extra Datei

```
01 @inproceedings{kahn1999next,  
02   author = {Kahn, J. M. and Katz, R. H. and  
03             Pister, K. S. J.},  
04   title = {Next Century Challenges: Mobile Networking  
05             for 'Smart Dust'},  
06   booktitle = {Proceedings of the 5th Annual International  
07                 Conference on Mobile Computing and  
08                 Networking (MobiCom)},  
09   year = {1999},  
10   pages = {271--278},  
11 }
```

■ Kompilieren per biber references.bib

- Referenzieren im Text

```
01 Kahn et al.~forged the term \emph{Smart Dust}~\cite{
    kahn1999next}.
```

Kahn et al. forged the term *Smart Dust* [**kahn1999next**].

- Einbinden per `\bibliography`

LaTeX Beamer

■ Neue Folie

```
01 \begin{frame}{Folienüberschrift}
02   Hallo Welt
03 \end{ frame}
```

■ Inhaltsverzeichnis

```
01 \begin{frame}{Folienüberschrift}
02   \tableofcontents
03 \end{ frame}
```

LaTeX

LaTeXBeamer

Versionskontrollsysteme

Zusammenfassung

■ Zwei benachbarte Boxen

```
01 \begin{minipage}[t]{0.45\textwidth}
02 Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam
    nonumy eirmod tempor invidunt ut labore et dolore magna
    aliquyam erat, sed diam voluptua.
03 \end{minipage}
04
05 \begin{minipage}[t]{0.45\textwidth}
06   \centering
07   \begin{itemize}
08     \item Punkt 1
09     \item Punkt 2
10   \end{itemize}
11
12   \includegraphics[width=0.8\textwidth]{img/git-logo}
13 \end{minipage}
```

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

- Punkt 1
- Punkt 2



Absolute Positionierung

- Manchmal ist es nötig ein {Text,Bild,...}-fragment absolut zu positionieren

```
01 %\begin{textblock*}{width} (x,y)  
02 \begin{textblock*}{0.25\textwidth}(5cm, 6.5cm)  
03 \includegraphics[width=\textwidth,keepaspectratio]{img/git-  
logo}  
04 \end{textblock*}
```



- Manchmal ist es nötig ein {Text,Bild,...}-fragment absolut zu positionieren

```
01 %\begin{textblock*}{width} (x,y)
02 \begin{textblock*}{0.25\textwidth}(5cm, 6.5cm)
03 \includegraphics[width=\textwidth,keepaspectratio]{img/git-
logo}
04 \end{textblock*}
```


- Manchmal ist es nötig ein {Text, Bild,...}-fragment absolut zu positionieren



```
01 %\begin{textblock*}{width} (x,y)
02 \begin{textblock*}{0.25\textwidth}(5cm, 6.5cm)
03 \includegraphics[width=\textwidth,keepaspectratio]{img/git-
    logo}
04 \end{textblock*}
```

```
01 \begin{itemize}
02   \item Punkt 1
03   \visible<2-> {
04     \item Punkt 2
05   }
06   \item Punkt 3
07 \end{itemize}
```

- Punkt 1
- Punkt 3

```
01 \begin{itemize}
02   \item Punkt 1
03   \only<2-> {
04     \item Punkt 2
05   }
06   \item Punkt 3
07 \end{itemize}
```

- Punkt 1
- Punkt 3

```
01 \begin{itemize}
02   \item Punkt 1
03   \visible<2-> {
04     \item Punkt 2
05   }
06   \item Punkt 3
07 \end{itemize}
```

- Punkt 1
- Punkt 2
- Punkt 3

```
01 \begin{itemize}
02   \item Punkt 1
03   \only<2-> {
04     \item Punkt 2
05   }
06   \item Punkt 3
07 \end{itemize}
```

- Punkt 1
- Punkt 2
- Punkt 3

Versionskontrollsysteme

Typische Aufgaben eines Versionskontrollsystems sind:

- Transportmedium
- Sichern von alten Zuständen
- Zusammenführung von parallelen Entwicklungen

Idealerweise zusätzlich:

- Unabhängige Entwicklung ohne zentrale Infrastruktur

Typische Aufgaben eines Versionskontrollsystems sind:

- Transportmedium
- Sichern von alten Zuständen
- Zusammenführung von parallelen Entwicklungen

Idealerweise zusätzlich:

- Unabhängige Entwicklung ohne zentrale Infrastruktur



- Jeder Studierende kann eigene Repositories im GitLab anlegen
- Login per SSO des RRZE
- Verhindert effektiv verlorene Ausarbeiten und Vorträge

→ <https://gitlab.cs.fau.de>

Wichtige Git Kommandos zum Austauschen von Code (1)

- Initialisieren einen Repos im aktuellen Verzeichnis

```
01 git init
```

- Initiales *Klonen* der Quellen

```
01 git clone https://gitlab.cs.fau.de/herzog/akss_ws18.git  
02 git clone git@gitlab.cs.fau.de:herzog/akss_ws18.git
```

- Hinzufügen einer neuen Datei zur Menge der von git versionierten Dateien

```
01 git add Datei
```


Wichtige Git Kommandos zum Austauschen von Code (2)

- Markieren einer versionierten Datei als Kandidat für den nächsten commit („staging“)

```
01 git add Datei
```

- Anzeige der Differenzen zum Vorgänger (bzw. Anzeige des vorbereiteten [„staged“] commits)

```
01 git diff
02 git diff [--staged|--cached]
```

- Dateizustände (neu, unbekannt, geändert, staged) anzeigen

```
01 git status
```

Wichtige Git Kommandos zum Austauschen von Code (3)

- Einspielen von eigenen Änderungen der Datei `text.txt` oder aller Änderungen

```
01 git commit text.txt
02 git commit -a
```

- Liste von Commits betrachten

```
01 git log
02 git log --graph
03 git log --pretty=[oneline|short|full|fuller]
```

- Den neusten bzw. einen bestimmten Commit untersuchen

```
01 git show
02 git show d4199363f9eeb542a6a2f3a590c28f55aef4f442
```

Wichtige Git Kommandos zum Austauschen von Code (4)

- Einspielen von entfernten Änderungen

```
01 git pull
```

- Veröffentlichen von lokalen Änderungen

```
01 git push
```

```
01 gitg  
02 gitk  
03 tig  
04  
05 git gui
```

- <https://gitlab.cs.fau.de>

Zusammenfassung

- Wissenschaftliches Schreiben mit \LaTeX
 - \LaTeX ist eine Textauszeichnungssprache
 - biblatex zur Literaturverwaltung
 - Unter anderem auch geeignet für die Erstellung von Präsentationen
- Versionsverwaltung mit Git
 - Verteiltes Versionskontrollsystem
 - Vereinfacht kollaboratives Arbeiten