

Seminar Konzepte von Betriebssystemkonzepte (KvBK)

Arbeitstechnik

24. April 2018

Benedict Herzog, Timo Hönig

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



Lehrstuhl für Verteilte Systeme
und Betriebssysteme



FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

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





Git wurde 2005 von Linus Torvalds zur Unterstützung für die Linux Kernel Entwicklung geschrieben. Es sind viele Erfahrungen im Umgang mit großen Patchmengen und das Vorgängersystem *bitkeeper*.

Es unterstützt:

- Dezentrale, parallele Entwicklung
- Koordinierung von Hunderten von Entwicklern
- Visualisierung von Entwicklungszweigen



- Initialisieren einen Repos im aktuellen Verzeichnis

```
01 git init
```

- Initiales Klonen der Quellen

```
01 git clone https://gitlab.cs.fau.de/herzog/kvbk_ss18.git
```

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

```
01 git commit text.txt
```

```
02 git commit -a
```



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

```
01 git add Datei
```

- 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
```

- Die neuste Änderung untersuchen

```
01 git show
```

- Einspielen von entfernten Änderungen

```
01 git pull
```




- Unterschiede zwischen lokalem und entferntem Branch untersuchen

```
01 git log ..origin/master  
02 git log -p ..origin/master
```



```
01 gitg
02 gitk
03 tig
04
05 git gui
06
07 # globales git difftool
08 git-meld
```

■ <https://gitlab.cs.fau.de>

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**

■ Kursiv

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

→ Hallo *Welt*



■ Festbreite

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

→ Hallo Welt

■ Text hervorheben

```
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]{fig/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



■ 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 }
```

■ Compilieren per bibtex 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 [1].



- Einbinden per `\bibliography`



J. M. Kahn, R. H. Katz, and K. S. J. Pister.

Next century challenges: Mobile networking for 'smart dust'.

In Proceedings of the 5th Annual International Conference on Mobile Computing and Networking (MobiCom), pages 271–278, 1999.

LaTeX Beamer



■ Neue Folie

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

■ Inhaltsverzeichnis

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

Versionskontrollsysteme

LaTeX

LaTeXBeamer



■ 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]{fig/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





```
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